UNIVERSITÀ DEGLI STUDI DI PISA

Facoltà di Ingegneria

CORSO DI LAUREA SPECIALISTICA IN INGEGNERIA
DELLE TELECOMUNICAZIONI

Tesi di laurea specialistica

Separazione di componenti dipendenti con applicazioni in astrofisica

Relatori:
Prof. Ercan Kuruoglu
Prof. Maria Sabrina Greco

Tesi di Laurea di:
Stefano Fortunati, 253100

Anno Accademico 2007-2008

Indice

Capitolo 1

Introduzione

In questa tesi ci occuperemo del problema della *separazione delle componenti*. Questo è un problema di grande rilevanza nel signal processing in quanto presenta numerose applicazioni. Un classico esempio è il *cocktail party problem*: supponiamo di essere in una stanza con delle persone che parlano simultaneamente. Nella stanza ci sono diversi microfoni piazzati a distanze diverse. Il segnale ricevuto da ogni microfono sarà una combinazione lineare delle voci delle varie persone. L'obiettivo della separazione sarà quello di recuperare i singoli segnali vocali partendo solo dalla conoscenza dei dati ricevuti dai microfoni. Questo problema può essere esteso a situazioni più generali che vanno sotto il nome di *blind source separation* ([1],[2],[3]). Le applicazioni vanno dalla separazione di sorgenti musicali ad alta definizione all'eliminazione dell'eco.

Un recente campo di utilizzo degli algoritmi di separazione si può trovare nell'elaborazione di immagini SAR (*Synthetic Aperture Radar*) e nel *remote sensing* ([4]). Le ragioni principali sono tre: l'estrazione delle feature, la riduzione del rumore e il data fusion. Possiamo infatti usare un algoritmo di separazione per estrarre le diverse feature dai dati SAR nella fase di classificazione (*estrazione delle feature*). I dati SAR possono essere considerati come una mistura di bersagli radar più una certa quantità di rumore. Un algoritmo di separazione, oltre che per dividere i vari bersagli, può essere utilizzato per eliminare il rumore (*riduzione del rumore*). Infine, un algoritmo di separazione può essere utile per eliminare dai dati la ridondanza prodotta dalle misure radar. Quando un certo numero di sensori osservano la stessa area, i

dati acquisiti hanno una notevole quantità di informazioni in comune. I dati ricevuti dai vari sensori potrebbero essere fusi insieme secondo un criterio che elimini questa ridondanza (*data fusion*).

Un altro campo di applicazione della separazione delle componenti è la medicina ([5]). Diamo una lista di esempi in campo medico:

- **Magnetoencefalogramma (MEG)** Il MEG è un esame medico che serve a misurare il campo magnetico prodotto dall'attività elettrica del cervello ([6]). E' estremamente importante separare i dati provenienti dalle varie regioni del cervello (le sorgenti) per studiarne le peculiarità o scoprire eventuali malattie.
- **Functional Magnetic Resonance Imaging (fMRI)** La fMRI è una tecnica simile alla precedente che misura la variazione di campo magnetico dovuta alle attività neurali ([7], [8]). La separazione delle componenti può essere usata per mappare le varie zone cerebrali a seconda della loro specifica funzione.
- **Elettrocardiogramma fetale (Fetal ECG)** Per fare analisi cardiache del feto è necessario misurarne l'elettrocardiogramma. Questo segnale però risulta molto più debole rispetto a quello proveniente dal cuore materno. Un possibile approccio a questo problema può essere quello di separare i due segnali tramite algoritmi di separazione ([5]).
- **Magnetogastrografica (MGG)** Lo stomaco, a causa della sua attività meccanica, genera segnali elettrici che possono essere studiati per rivelare eventuali patologie. Il problema però è che ogni muscolo emette segnali elettromagnetici. Isolare i vari contributi (ad esempio cuore, stomaco, muscoli motori ecc) è un problema difficile. Anche in questo caso si possono applicare algoritmi di separazione ([9]).

Altre applicazioni si trovano in ambito finanziario ([10], [11]). Si definiscono *serie finanziarie* quei dati che caratterizzano scambi monetari, flussi di importazioni ed esportazioni o il variare delle giacenze di un magazzino. L'approccio statistico a questi problemi può essere molto utile per approntare una strategia di investimento. Si possono, ad esempio, sviluppare modelli che ipotizzano

un probabile andamento futuro del mercato partendo dai dati già acquisiti. In questo contesto la separazione delle componenti può essere applicata per differenziare flussi finanziari diversi appartenenti allo stesso insieme di dati. Facciamo un esempio molto semplice: supponiamo che i nostri dati rappresentino gli incassi di una grande catena di negozi. Per migliorare le strategie di marketing o riorganizzare i punti vendita, servirebbe conoscere i flussi di incassi provenienti dai singoli negozi o dai singoli settori di prodotti. Anche questo problema può essere affrontato mediante algoritmi di separazione.

L'applicazione di cui ci occuperemo in questa tesi è quella relativa all'astrofisica. Negli ultimi anni sono state approntate diverse missioni spaziali che miravano a raccogliere dati utili per capire meglio l'origine e lo stato attuale del nostro Universo. I dati raccolti da queste missioni sono una combinazione lineare dei segnali provenienti dalle varie sorgenti. Prima di poterli studiare, bisogna quindi isolare i vari contributi. Il problema è quindi un problema di separazione.

Tutte le applicazioni appena descritte sono accomunate dallo stesso modello matematico. I dati raccolti infatti, siano essi di natura audio, medica, finanziaria o astrofisica, possono essere visti come una combinazione lineare delle varie sorgenti. Formalmente il problema può essere esposto nel seguente modo: siano $x \in \mathbb{R}^m$ e $s \in \mathbb{R}^{m'}$ due vettori aleatori legati dalla relazione $x = As$, dove A, detta *mixing matrix*, è una matrice di dimensioni $m \times m'$. Il nostro obiettivo sarà quello di stimare la *demixing matrix* W tale che $Wx = s$. Restringeremo la nostra analisi al caso in cui $m' \leq m$, cioè quando ci sono più sensori che sorgenti. Molti degli algoritmi proposti per la soluzione di questo problema non assumono nessuna informazione a priori sulla distribuzione statistica o sul numero delle sorgenti (in questi casi la separazione è detta "blind"). Il principale algoritmo utilizzato per la separazione *blind* delle componenti è quello ICA: *Indipendent Component Analysis*. L'algoritmo ICA impone come ipotesi l'indipendenza statistica delle sorgenti. Nella maggior parte delle applicazioni però, questa ipotesi non è soddisfatta. Nel caso dei dati medici, soprattutto per quanto riguarda l'fMRI, esiste una forte dipendenza tra le varie sorgenti. Se questa dipendenza non viene presa in considerazione le prestazioni della separazione possono risultare particolarmente basse. Anche nel caso della separazione

di tracce audio provenienti da varie sorgenti musicali, l'ipotesi di indipendenza è violata. Gli strumenti infatti, non suonano indipendentemente l'uno dall'altro, ma seguono un precisa composizione. Nell'applicazione che tratteremo in questa tesi, il problema astrofisico, le varie sorgenti non sono mutualmente indipendenti, ma presentano una struttura di dipendenza prevista dai modelli fisici esistenti. L'algoritmo che studieremo in dettaglio in questa tesi abbandona l'ipotesi di indipendenza usata dall'ICA e ammette per le sorgenti una struttura di dipendenza ad albero. Questo algoritmo va sotto il nome di TCA: *Tree-dependent component analysis*.

Nella prima parte della tesi daremo una veloce descrizione di alcuni algoritmi per la separazione e studieremo in dettaglio quello TCA. Nella seconda parte verrà discusso il problema astrofisico, dando una descrizione delle varie sorgenti e accennando alla missione dell'ESA Planck. Verranno poi mostrati i risultati ottenuti con dati sintetici sia simulati in Matlab sia astrofisici. Tratteremo anche del problema del rumore degli strumenti e descriveremo i tentativi fatti per eliminarlo. In ultimo applicheremo l'algoritmo TCA a dati reali provenienti dalla missione WMAP e trarremo le conclusioni.

Struttura della tesi:

Capitolo 1 Introduzione.

Capitolo 2 Verranno brevemente presentati diversi algoritmi per la separazione delle componenti (ICA (Independent Component Analysis), Multidimensional ICA, Topographic ICA, CorCA), e verrà analizzato in dettaglio l'algoritmo TCA (Tree-dependent Component Analysis).

Capitolo 3 Verrà analizzato il problema della separazione delle componenti in ambito astrofisico. Descriveremo brevemente gli obiettivi delle varie missioni spaziali approntate per raccogliere dati e analizzeremo i tipi di sorgenti rilevanti per la cosmologia.

Capitolo 4 Testeremo l'algoritmo TCA con diversi tipi di dati. In primo luogo useremo dei dati generati in Matlab, poi prenderemo in considerazione dati astrofisici sintetici e in ultimo useremo dei dati reali ottenuti dalla missione WMAP. Verrà descritto il problema del rumore. Faremo inoltre il

confronto con gli altri algoritmi di separazione descritti nel Capitolo 2. Infine trarremo le conclusioni.

Capitolo 2

Algoritmi per la separazione delle componenti

In questo primo capitolo presenteremo brevemente vari algoritmi che sono stati proposti per risolvere il problema della separazione delle componenti. Nell'ultima sezione analizzeremo con attenzione e in tutti i suoi aspetti l'algoritmo TCA che è l'oggetto di questa tesi.

2.1 L'algoritmo ICA: *Indipendent Component Analysis*

Il modello usato dall'algoritmo ICA è quello già descritto nell'introduzione ([12]). Siano $x \in \mathbb{R}^m$ e $s \in \mathbb{R}^m$ due vettori aleatori legati dalla relazione $x = As$, dove A è una matrice non singolare di dimensioni $m \times m$. x rappresenta il vettore delle osservazioni mentre s quello delle sorgenti. Per l'algoritmo ICA si assume che le sorgenti $s_1, \ldots, s_m$ siano *statisticamente indipendenti*. L'obiettivo sarà quello di trovare una matrice W, detta *demixing matrix* tale che le componenti del vettore $s' = Wx$ risultino "il più indipendenti possibile". Vedremo che tutti i possibili metodi di stima di W si basano sull'assunzione di indipendenza delle componenti di s. Prima di passare a descrivere questi metodi è importante fare una precisazione: il vettore s può contenere al

più una sola componente con distribuzione Gaussiana. Se ci fosse più di una componente Gaussiana la demixing matrix potrebbe essere stimata solo a meno di trasformazioni ortogonali. In pratica questo significa che non si può in nessun modo risalire a W. Una dimostrazione di questo fatto si può trovare in ([13]).

Stima di W basata sulla non-Gaussianità

Discutiamo ora un primo metodo di stima della demixing matrix. Questo metodo sfrutta un risultato fondamentale della teoria della probabilità: il *teorema del limite centrale*. Questo teorema afferma che la somma (al limite di infiniti termini) di variabili aleatorie indipendenti, tende ad una distribuzione Gaussiana. In altre parole, la somma di due variabili indipendenti è più vicina ad una Gaussiana di quanto non lo fossero le due variabili di partenza. Torniamo ora al problema ICA. Come abbiamo detto le componenti s_i di s sono assunte indipendenti. Ricordando il modello, $x = As$, facciamo questo cambio di variabile: $y = w^t x$, dove w è un generico vettore reale. Se ora pensiamo a w^t come ad una riga della demixing matrix W, potremmo affermare che y sia una delle componenti di s. Quello che ci chiediamo ora è: come possiamo sfruttare il teorema del limite centrale per stimare w^t in modo che y sia una componente di s? Cominciamo facendo un cambio di variabile. Poniamo

$$z = A^t w \tag{2.1}$$

e quindi

$$y = w^t x = w^t A s = z^t s \tag{2.2}$$

Si nota subito che la variabile aleatoria y è una combinazione lineare delle componenti di s, in formule $y = \sum_i z_i s_i$. Ora, dal teorema del limite centrale, si può affermare che y sia più Gaussiana di ogni componente s_i. Se z avesse una sola componente, ad esempio z_i, diversa da 0, allora y sarebbe proporzionale a s_i. Questo ci suggerisce che per stimare W, potremmo massimizzare la non-Gaussianità di $w^t x$. Il vettore w così ottenuto sarà infatti quello che, tramite il cambio di variabile in (2.1), darà un vettore z con una sola componente diversa da 0.

La misura più classica di non-Gaussianità è la kurtosis. Questa quantità è definita come:

$$kurt\,(y) = E\left\{y^4\right\} - 3\left(E\left\{y^2\right\}\right)^2 \tag{2.3}$$

Per variabili aleatorie Gaussiane la kurtosis è nulla. Si può quindi definire uno stimatore per W (in particolare per le sue righe) come:

$$\hat{w} = \arg\max_{w} \left\{ \left| kurt\left(w^t x\right) \right| \right\} \tag{2.4}$$

Stima di W a massima verosimiglianza

Un altro approccio per la stima della demixing matrix dell'algoritmo ICA è quella della stima a massima verosimiglianza. Il logaritmo della funzione di verosimiglianza del modello è stata calcolata ([14]) e vale:

$$L\left(W\right) = \sum_{n=1}^{N} \sum_{i=1}^{K} \log p_{s_i}\left(w_i^t x_n\right) + N \log\left|\det W\right| \tag{2.5}$$

dove p_{s_i} sono le densità di probabilità delle componenti s_i, w_i sono le righe di W e x_n con $n = 1, \ldots, N$ sono le realizzazioni di x.

Stima di W mediante l'informazione mutua

Una buona misura dell'indipendenza è l'informazione mutua. L'informazione mutua $I(s_1, \ldots, s_m)$ è uguale a 0 se e solo se le variabili $s_1, \ldots, s_m$ sono statisticamente indipendenti (guardare appendice A). Un modo per implementare questo algoritmo è quello di definire come contrast function (una funzione cioè che dipende solo dalle osservazioni e dalle grandezze da stimare) proprio l'informazione mutua e minimizzare questa grandezza rispetto a W. In maniera rigorosa possiamo definire la contrast function come:

$$J\left(x, W\right) = I\left(s'_1, \ldots, s'_m\right)$$

con $s' = Wx$. E quindi, per quanto detto sopra, la matrice W stimata con l'algoritmo ICA sarà:

$$\hat{W}_{ICA} = \arg\min_{W} J\left(x, W\right) = \arg\min_{W} I\left(s'_1, \ldots, s'_m\right)$$

Ambiguità dell'algoritmo ICA

Partendo dal modello $x = As$ è facile vedere che la stima della demixing matrix è affetta da due tipi di ambiguità.

1. Non è possibile determinare l'esatta ampiezza delle componenti s_i.

   La ragione di questo sta nel fatto che, essendo sia s che A sconosciuti, ogni eventuale fattore moltiplicativo presente su una sorgente s_i può essere cancellato dividendo per lo stesso numero la relativa colonna a_i della matrice A. In forma matriciale possiamo scrivere:

   $$x = As = \left(A\Lambda^{-1}\right)(\Lambda s) \tag{2.6}$$

   dove Λ è una matrice diagonale reale.

2. Non è possibile determinare l'ordine delle componenti

   Le componenti di s possono essere permutate in maniera qualsiasi a patto di effettuare la stessa permutazione sulle colonne di A. In forma matriciale:

   $$x = As = \left(AP^{-1}\right)(Ps)$$

In conclusione possiamo dire che, usando l'algoritmo ICA, la demixing matrix può essere stimata a meno di fattori di scala e della permutazione delle sue righe.

Come abbiamo già visto nell'introduzione, ci sono molte applicazioni in cui imporre l'indipendenza tra le sorgenti può risultare una condizione troppo forte. Da qui fino alla fine del capitolo, presenteremo altri algoritmi (studiando nel dettaglio l'algoritmo TCA) che rilassano l'ipotesi di indipendenza in vari modi. Alcuni di questi possono essere visti come una generalizzazione dell'algoritmo ICA (ad esempio *MICA* e *Topographic ICA*) mentre altri usano degli approcci totalmente diversi (*CorCA* e *TCA*).

2.2 L'algoritmo MICA: *Multidimensional ICA*

Vediamo ora un algoritmo che mira a estendere i concetti usati nell'algoritmo ICA. Cominciamo formulando in maniera diversa il modello ICA (l'articolo di riferimento è [15]). Definiamo *componenti* i vettori $x_p = s_p a_p$ dove s_p sono le sorgenti e a_p sono

le colonne della mixing matrix A. Il modello può essere scritto come:

$$x = As = \sum_{p=1}^{m} x_p$$

Notiamo che tutti i componenti x_p appartengono allo spazio monodimensionale generato dalla rispettiva colonna di A, in simboli $x_p \in Span\,(a_p)$. Definiamo ora la matrice "proiettore" su questi spazi come:

$$\Pi_p = \frac{a_p a_p^t}{\|a_p\|^2} \tag{2.7}$$

Dato infatti un vettore z, la sua proiezione ortogonale sullo $Span\,(a_p)$ (a meno di costanti) è data da: $\Pi_p z = \frac{a_p a_p^t}{\|a_p\|^2} z = \frac{\langle a_p, z\rangle}{\|a_p\|^2} a_p$ con $p \in \{1, \ldots, m\}$. Vediamo ora come la conoscenza dell'insieme dei proiettori $\mathcal{P} = \{\Pi_1, \ldots, \Pi_m\}$ sia sufficiente per separare le componenti. Come prima cosa dobbiamo ortogonalizzare la base formata dalle colonne di A. Questo ci consentirà di scomporre x nelle sue proiezioni ortogonali sulle componenti di questa base. Si può dimostrare che, definendo $\tilde{\Pi}_p = \Pi_p \left(\sum_{q=1}^{m} \Pi_q \right)$, possiamo recuperare le componenti, infatti:

$$x_p = \tilde{\Pi}_p x = \Pi_p \left(\sum_{q=1}^{m} \Pi_q \right)^{-1} x \tag{2.8}$$

Questa formulazione *geometrica* dell'algoritmo ICA può essere estesa al caso multidimensionale. Diamo ora alcune definizioni utili:

Definizione 2.1. *Siano $E_1, \ldots, E_c$ sottospazi lineari di $\mathbb{R}^m$. Essi sono detti linearmente indipendenti se ogni vettore x appartenente a $E_1 \oplus \cdots \oplus E_c$ ammette una decomposizione unica $x = \sum_{p=1}^{c} x_p$ con $x_p \in E_p$ per $1 \leq p \leq c$. In questo caso i vettori $x_1, \ldots, x_c$ sono detti componenti lineari.*

Definizione 2.2. *Un vettore aleatorio m-dimensionale x ammette una decomposizione MICA in c componenti $\{x_1, \ldots, x_c\}$ se esistono c sottospazi lineari di $\mathbb{R}^m$ nei quali le componenti lineari di x sono statisticamente indipendenti.*

Per capire il significato di questa definizione, riconsideriamo il modello ICA nel modo clasico. Siano $s_1, \ldots, s_c$ c vettori aleatori indipendenti di dimensioni $n_1, \ldots, n_c$ e sia $s = (s_1^t, \ldots, s_c^t)^t$. Siano ora $A_1, \ldots, A_c$ c matrici di dimensioni $m \times n_1, \ldots, m \times n_c$ con $n_1 + \cdots + n_c = m$ tali che $A = (A_1, \ldots, A_c)$ con $\det(A) \neq 0$. Secondo il modello ICA avremo che $x = As$. Quindi, il vettore x ammette una decomposizione MICA negli spazi $E_1, \ldots, E_c$ dove $E_p = Span(A_p)$ in accordo con le definizioni 1 e 2. Un generico proiettore ortogonale nello spazio E_p è dato da:

$$\Pi_p = A_p \left(A_p^t A_p\right)^{-1} A_p^t$$

che rappresenta l'equivalente multidimensionale della (2.7). L'equazione (2.8) mantiene la stessa forma. Le componenti lineari x_p restano così unicamente determinate dall'insieme dei proiettori.

Problemi irrisolti dell'algoritmo MICA

Quanto finora detto è tutto formalmente corretto. In pratica però i problemi irrisolti sono molti. L'algoritmo MICA può essere usato solo per un post-processing empirico della stima ICA. Per vedere se un dato vettore aleatorio ammette una decomposizione MICA si deve procedere in due passi:

1. Stimare tramite l'algoritmo ICA la demixing matrix W e quindi le varie sorgenti supposte indipendenti.
2. Determinare quali sorgenti sono indipendenti e quali invece possono essere raggruppate insieme e viste come un'unica componente multidimensionale.

Il secondo passo però viene fatto in maniera del tutto empirica. Allo stato attuale questo algoritmo non migliora le prestazioni per la separazione delle componenti, può essere usato soltanto per un'analisi della stima ICA. L'algoritmo TCA che presenteremo più avanti cerca di generalizzare l'idea del MICA e la usa per migliorare le prestazioni della separazione.

2.3 L'algoritmo TICA: *Topographic ICA*

Passiamo ora ad analizzare brevemente una possibile modifica dell'algoritmo ICA che possa in qualche modo tener conto della

dipendenza tra le componenti di s. Per far questo introduciamo il concetto di *mappa topografica*. Potremmo pensare di posizionare in una mappa le componenti s_i ordinandole a seconda della loro reciproca dipendenza: le componenti con una forte dipendenza saranno più vicine, mentre quelle debolmente dipendenti saranno più lontane. Da questa idea è stato sviluppato l'algoritmo *Topographic ICA* (TICA) ([16]). La prima cosa da stabilire è il tipo di dipendenza da usare come distanza tra le componenti. Per questo scopo, l'algoritmo TICA usa una correlazione di ordine superiore al secondo, detta correlazione di energie. Questa è definita come segue:

$$\operatorname{cov}\left\{s_i^2, s_j^2\right\} = E\left\{s_i^2 s_j^2\right\} - E\left\{s_i^2\right\} E\left\{s_j^2\right\} \tag{2.9}$$

Modello del segnale

Il modello delle osservazioni è lo stesso usato dall'algoritmo ICA, cioè $x = As$. Questa volta però, al fine di stimare la demixing matrix W, faremo delle ipotesi sul vettore delle sorgenti s. L'idea è quella di considerare la varianza σ_i^2 della componente s_i non come una costante, ma come una variabile aleatoria. La dipendenza tra le varie s_i sarà contenuta nella loro varianza σ_i^2, in altre parole le componenti s_i risultano indipendenti data la loro varianza.

Definiamo ora una *funzione di vicinanza* $h(i, j)$ che esprime la distanza tra la i-esima e la j-esima componente di s. Questa funzione deve soddisfare due importanti proprietà:

- $h(i, j) = h(j, i)$, proprietà di simmetria
- $h(i, i) = costante$ per qualsiasi i

Un semplice esempio di questa funzione può essere:

$$h\left(i, j\right) = \begin{cases} 1 & |i - j| \leqslant m \\ 0 & altrimenti \end{cases} \tag{2.10}$$

Tramite questa funzione di vicinanza possiamo modellare la deviazione standard come segue:

$$\sigma_i = \phi\left(\sum_{k=1}^{n} h\left(i, k\right) u_k\right) \tag{2.11}$$

dove u_i sono delle variabili aleatorie indipendenti e ϕ è una generica funzione non lineare. Possiamo infine esibire il modello della generica componente s_i:

$$s_i = \phi \left(\sum_{k=1}^{n} h\left(i,k\right) u_k \right) z_i \tag{2.12}$$

dove z_i sono delle variabili aleatorie con varianza unitaria e mutualmente indipendenti da u_k, con $k = 1, \ldots, n$. Questo particolare modello ci permette di esprimere in funzione delle sole σ_i^2 la distanza tra le componenti. Avremo infatti:

$$\begin{aligned} d\left(s_i, s_2\right) &= \operatorname{cov}\left\{s_i^2, s_j^2\right\} = \\ &= E\left\{\sigma_i^2 z_i^2 \sigma_j^2 z_j^2\right\} - E\left\{\sigma_i^2 z_i^2\right\} E\left\{\sigma_j^2 z_j^2\right\} = \\ &= E\left\{\sigma_i^2 \sigma_j^2\right\} - E\left\{\sigma_i^2\right\} E\left\{\sigma_j^2\right\} \end{aligned} \tag{2.13}$$

Ulteriori proprietà di questo modello possono essere trovate in ([16]).

Stima della demixing matrix

Usando il modello di s appena descritto dobbiamo stimare la demixing matrix W. L'approccio usato consiste nel fare una stima a massima verosimiglianza ([14]) simile a quella usata nell'algoritmo ICA in (2.5). La funzione di verosimiglianza in questo caso assume la forma:

$$L\left(W\right) = \prod_{n=1}^{N} \int \prod_{i} p_{s_i} \left(\frac{w_i^t x_n}{\phi\left(\sum_k h\left(i,j\right) u_k\right)} \right) \frac{p_{u_i}\left(u_i\right)}{\phi\left(\sum_k h\left(i,j\right) u_k\right)} \left|\det W\right| du \tag{2.14}$$

dove p_{s_i} e p_{u_i} rappresentano le densità di probabilità di s_i e u_i rispettivamente, x_n con $n = 1, \ldots, N$ sono le realizzazioni di x e w_i indica l'i-esima riga di W presa come vettore colonna. L'integrale nella (2.14) non può essere scritto in forma chiusa. Bisogna quindi trovare una valida approssimazione per poi applicare l'algoritmo di minimizzazione.

2.4 Separazione di sorgenti correlate: *CorCA*

Come abbiamo già detto, l'algoritmo ICA richiede la mutua indipendenza tra le varie sorgenti. Questa ipotesi però non risulta sempre soddifatta. In astrofisica, ad esempio, è noto che alcune sorgenti sono dipendenti tra loro e indipendenti da altre. Per migliorare le prestazioni della separazione bisognerebbe prendere in considerazione modelli più generali che ammettono delle dipendenze tra le sorgenti. Un'altra limitazione dell'algoritmo ICA è quella di non consentire l'utilizzo di eventuali informazioni a priori che potrebbero servire a migliorare la stima. Questo modo di operare è detto *blind source separation* in quanto è in grado di stimare la demixing matrix senza conoscere nulla delle sorgenti. L'algoritmo a cui accenniamo in questa sezione si distingue in almeno due punti dall'algoritmo ICA, infatti:

- Sfrutta eventuali informazioni a priori
- Ammette una dipendenza lineare fra le varie sorgenti

L'algoritmo CorCA ([17]) sfrutta proprio la dipendenza lineare e le matrici di covarianza per stimare la demixing matrix.

Descrizione dell'algoritmo

L'algoritmo è stato scritto per risolvere il problema della separazione di sorgenti astrofisiche, il modello del segnale che adotteremo sarà conforme a questo problema.

Indichiamo con $x(\xi,\eta)$ il vettore delle osservazioni di dimensione m (m indica il numero di canali), con $s(\xi,\eta)$ il vettore delle sorgenti di dimensione n e con $n(\xi,\eta)$ il rumore dovuto agli strumenti. Gli indici (ξ,η) indicano la posizione del pixel. Il modello sarà quindi:

$$x(\xi,\eta) = As(\xi,\eta) + n(\xi,\eta) \tag{2.15}$$

dove A è una matrice reale di dimensioni $m \times n$ (torneremo su questo modello più avanti nella tesi). Assumeremo che tutti i processi $x(\xi,\eta)$, $s(\xi,\eta)$ e $n(\xi,\eta)$ siano stazionari. Del processo di rumore possiamo conoscere tutte le statistiche e in particolare la matrice di covarianza. L'ipotesi di stazionarietà in molti casi

risulta essere un'approssimazione, ad esempio il rumore d'antenna risulta essere fortemente non stazionario. Assumiamo anche di conoscere a priori alcuni coefficienti della matrice A. Questo, oltre a migliorare la precisione della separazione, fa diminuire la complessità dell'algoritmo, riducendo il numero di equazioni necessarie per la stima. L'informazione a priori deriva dalla conoscenza del sistema che si sta osservando.

La matrice di covarianza dei processi $s(\xi,\eta)$ e $n(\xi,\eta)$ è definita come:

$$C_s(\tau,\psi) = E\left\{(s(\xi,\eta)-\mu_s)(s(\xi+\tau,\eta+\psi)-\mu_s)^t\right\}$$

$$C_n(\tau,\psi) = E\left\{(n(\xi,\eta)-\mu_n)(n(\xi+\tau,\eta+\psi)-\mu_n)^t\right\}$$

rispettivamente. I vettori μ_s e μ_n sono i rispettivi valori medi. Il processo di rumore è assunto indipendente dalle sorgenti $s(\xi,\eta)$, bianco e a valor medio nullo. Risulta quindi che per $(\tau,\psi) = (0,0)$, $C_n(0,0)$ è una matrice diagonale che contiene la varianza delle componenti di rumore su ciascun canale, mentre per $(\tau,\psi) \neq (0,0)$, $C_n(\tau,\psi)$ è una matrice nulla.

Occupiamoci ora della matrice di covarianza dei dati. Sfuttando la definizione e sostituendo poi il modello (2.15) risulta essere:

$$\begin{aligned} C_x(\tau,\psi) &= E\left\{(x(\xi,\eta)-\mu_x)(x(\xi+\tau,\eta+\psi)-\mu_x)^t\right\} \\ &= AC_s(\tau,\psi)A^t + C_n(\tau,\psi) \end{aligned}$$

La $C_x(\tau,\psi)$ può essere stimata dalle osservazioni nel seguente modo:

$$\hat{C}_x(\tau,\psi) = \frac{1}{N_p}\sum_{\xi,\eta}(x(\xi,\eta)-\mu_x)(x(\xi+\tau,\eta+\psi)-\mu_x)^t$$

dove N_p indica il numero di pixel.

Ci proponiamo ora di stimare A e $C_s(\tau,\psi)$ dalle matrici note $\hat{C}_x(\tau,\psi)$ e $C_n(\tau,\psi)$. In questa stima vogliamo anche tenere conto delle eventuali informazioni a priori sui coefficienti di A e di $C_s(\tau,\psi)$. Lo stimatore proposto è il seguente:

$$(\Gamma,\Sigma(.,.)) = \arg\min\sum_{\tau,\psi}\left\|A(\Gamma)C_s(\Sigma(\tau,\psi))A(\Gamma)^t - \hat{C}_x(\tau,\psi) + C_n(\tau,\psi)\right\|$$

dove Γ è il vettore dei coefficienti incogniti di A e $\Sigma(.,.)$ è il vettore dei coefficienti incogniti di $C_s(\tau, \psi)$ per ogni possibile coppia (τ, ψ). La norma usata è quella di Frobenius.

Nei problemi in cui è presente anche il rumore, la stima della demixing matrix è solo il primo passo. Il passo successivo è quello di stimare il vettore delle sorgenti s. Indichiamo con W l'inversa di Moore-Penrose di A definita come:

$$W = \left(A^t A\right)^{-1} A^t$$

e moltiplichiamo per W entrambi i membri della (2.15). Avremo:

$$Wx = s + Wn$$

Il termine Wn rappresenta un termine di rumore che di solito ha una potenza maggiore di quella di n. Bisognerà quindi applicare una qualche tecnica di filtraggio per eliminare questo disturbo. Il problema del termine di rumore Wn si presenta anche quando si usano altri tipi di algoritmi, ad esempio ICA o TCA. Torneremo più avanti su questo problema.

2.5 TCA: *Tree-dependent Component Analysis*

L'ipotesi dell'indipendenza delle sorgenti che è alla base dell'algoritmo ICA può risultare troppo stringente. In alcune applicazioni, come ad esempio quella della separazione di sorgenti astrofisiche, le sorgenti presentano una forte dipendenza che deve essere presa in considerazione per migliorare le prestazioni. Se quindi l'algoritmo ICA ci permette di trovare una trasformazione lineare W tale che le componenti del vettore $s = Wx$ risultino il più indipendenti possibile (data una misura d'indipendenza), quello che ci proponiamo ora è di generalizzare questa idea: cercheremo cioè una trasformazione lineare W tale che le componenti di $s = Wx$ possano essere modellate su una struttura ad albero (o per meglio dire, su un grafo) che caratterizzerà la struttura di dipendenza. La topologia dell'albero non deve essere fissata a priori. Questo metodo viene detto *tree-dependent component analysis* o TCA ([18]). Cominceremo la nostra analisi di questo algoritmo introducendo alcuni concetti di base sui *modelli grafici*.

2.5.1 Modelli grafici

Un modello grafico stabilisce una corrispondenza tra famiglie di funzioni di densità di probabilità e un grafo. Se $G(\mathcal{V}, \mathcal{E})$ è un grafo dove $\mathcal{V}$ è l'insieme dei vertici e $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ è l'insieme dei lati, possiamo associare a ogni nodo $v \in \mathcal{V}$ una variabile aleatoria x_v. La topologia del grafo caratterizza le relazioni esistenti tra le varie variabili aleatorie. Ogni lato presente nel grafo indicherà che i vertici ad esso collegati rappresentano due variabili aleatorie dipendenti. Se tutte le variabili fossero indipendenti (come nel modello ICA), il grafo corrispondente non avrà nessun lato. In generale, data una qualsiasi densità di probabilità è possibile associarle un grafo. Noi però non tratteremo il caso generale ma ci limiteremo al caso degli alberi (figura 2.1). Un albero $T(\mathcal{V}, \mathcal{E})$ è un grafo nel quale c'è al più un singolo percorso tra una particolare coppia di nodi. Notiamo che la definizione permette di non avere nessuna connessione tra due nodi. In particolare potremmo avere un albero formato da più componenti non connesse tra loro (in questo caso si parla di $foresta$). Se invece, dato un albero T, è sempre possibile trovare un percorso tra due nodi qualsiasi allora T è detto *spanning tree*. Per modellare una densità di probabilità su un albero definiamo delle funzioni (dette *potenziali*) $\psi_{uv}(x_u, x_v)$ e $\psi_u(x_u)$ per $(u, v) \in \mathcal{E}$ e $u \in \mathcal{V}$. Questi potenziali sono delle arbitrarie funzioni non-negative. La densità di probabilità congiunta relativa a un albero T è definita come ([19]):

$$p(x) = \frac{1}{D} \prod_{(u,v)\in\mathcal{E}} \psi_{uv}(x_u, x_v) \prod_{u\in\mathcal{V}} \psi_u(x_u) \tag{2.16}$$

dove D è un fattore di normalizzazione. Fissato un albero T e variando i potenziali, otteniamo tutta la famiglia di densità di probabilità che fattorizzano su T.

E' sempre possibile riscrivere la (2.16) in funzione delle densità di probabilità (ddp) marginali. La (2.16) diventerà quindi:

$$p(x) = \prod_{(u,v)\in\mathcal{E}} \frac{p_{uv}(x_u, x_v)}{p_u(x_u)\, p_v(x_v)} \prod_{u\in\mathcal{V}} p_u(x_u) \tag{2.17}$$

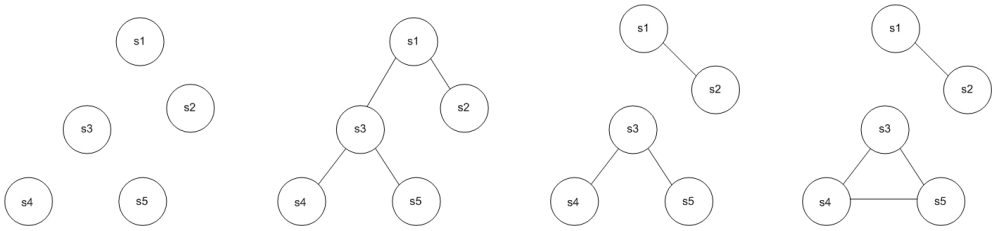


Figura 2.1: (a)topologia usata dall'algoritmo ICA, (b)esempio di spanning tree, (c)esempio di foresta, (d)esempio di cluster. Le topologie (a) (b) (c) sono ammesse dall'algoritmo TCA, la (d) no

2.5.2 Indipendenza condizionata

Discutiamo ora un'importante proprietà, molto semplice nel caso di alberi. Le variabili $(x_1, x_2, ..., x_m)$ fattorizzano su T se e solo se per ogni tre sottoinsiemi A, B e C di $\mathcal{V}$ tali che C separa A da B nel grafo, gli insiemi di variabili $x_A = \{x_i, i \in A\}$, $x_B = \{x_i, i \in B\}$ e $x_C = \{x_i, i \in C\}$ sono tali che x_A è indipendente da x_B dato x_C (indicheremo questo fatto con $A \perp\!\!\!\perp B \mid C$). Riprenderemo più avanti questa proprietà.

2.5.3 T-Mutual Information

Dato un albero T con insieme dei vertici $\mathcal{V} = \{1, \ldots, m\}$, indichiamo con $\mathcal{D}^T$ l'insieme delle densità di probabilità $q(x)$ che fattorizzano su T. Noi vogliamo modellare una generica $p(x)$ (dove le componenti di x non fattorizzano necessariamente in T) con una ddp appartenente a $\mathcal{D}^T$ minimizzando la divergenza KL tra le due distribuzioni. Vale il seguente teorema:

Teorema 2.1. *Dati un albero $T(\mathcal{V}, \mathcal{E})$ e una densità p, avremo per ogni $q \in \mathcal{D}^T$ la seguente espansione (detta pitagorica) della divergenza KL*

$$D(p \parallel q) = D(p \parallel p_T) + D(p_T \parallel q)$$

dove $p_T(x) = \prod_{(u,v)\in\mathcal{E}} \frac{p_{uv}(x_u, x_v)}{p_u(x_u)p_v(x_v)} \prod_{u\in\mathcal{V}} p_u(x_u)$. *Inoltre* $q = p_T$ *mini-*

mizza $D(p \parallel q)$ su $\mathcal{D}^T$. Avremo infine che:

$$\begin{aligned} I^T(x) &= \min_{q \in \mathcal{D}^T} D(p \parallel q) = D(p \parallel p_T) = \\ &= I(x_1, \ldots, x_m) - \sum_{(u,v) \in \mathcal{E}} I(x_u, x_v) \end{aligned} \tag{2.18}$$

$I^T(x)$ è detta T-mutual information.

Dimostrazione

Dato che q appartiene a $\mathcal{D}^T$ sarà della forma $q(x) = \prod_{(u,v)\in\mathcal{E}} \frac{q_{uv}(x_u,x_v)}{q_u(x_u)q_v(x_v)} \prod_{u\in\mathcal{V}} q_u(x_u)$.
Il primo passo della dimostrazione sarà mostrare che $\int p(x) \log q(x) dx = \int p_T(x) \log q(x)\, dx$.

$$\begin{aligned} &\int p(x) \log q(x) = \\ &= \int p(x_1, \ldots, x_m) \log \left(\prod_{(u,v)\in\mathcal{E}} \frac{q_{uv}(x_u, x_v)}{q_u(x_u)\, q_v(x_v)} \prod_{u\in\mathcal{V}} q_u(x_u) \right) dx_1 \ldots dx_m = \\ &= \int p(x_1, \ldots, x_m) \left(\sum_{(u,v)\in\mathcal{E}} \log \frac{q_{uv}(x_u, x_v)}{q_u(x_u)\, q_v(x_v)} + \sum_{u\in\mathcal{V}} \log q_u(x_u) \right) dx_1 \ldots dx_m = \\ &= \sum_{(u,v)\in\mathcal{E}} \int p_{uv}(x_u, x_v) \log \frac{q_{uv}(x_u, x_v)}{q_u(x_u)\, q_v(x_v)} dx_u dx_v + \sum_{u\in\mathcal{V}} \int p_u(x_u) \log q_u(x_u)\, dx_u \end{aligned}$$

Vediamo che nell'ultimo passaggio sono coinvolte solo le densità marginali di $p(x)$ relative alle variabili sui lati e sui vertici dell'albero. Queste densità sono esattamente quelle tramite le quali abbiamo definito la $p_T(x)$. Possiamo quindi affermare che $\int p(x) \log q(x) dx = \int p_T(x) \log q(x)\, dx$.

Dimostriamo ora, sfruttando l'uguaglianza appena trovata, che

vale l'espansione pitagorica per la divergenza KL. Abbiamo che:

$$\begin{aligned} D(p \parallel q) &= \int p(x) \log p(x)\, dx - \int p(x) \log q(x)\, dx = \\ &= \int p(x) \log p(x)\, dx - \int p_T(x) \log q(x)\, dx = \\ &= \int p(x) \log p(x)\, dx - \int p_T(x) \log p_T(x)\, dx + \\ &+ \int p_T(x) \log p_T(x)\, dx - \int p_T(x) \log q(x)\, dx = \\ &= \int p(x) \log p(x)\, dx - \int p(x) \log p_T(x)\, dx + \\ &+ \int p_T(x) \log p_T(x)\, dx - \int p_T(x) \log q(x)\, dx = \\ &= D(p \parallel p_T) + D(p_T \parallel q) \end{aligned} \quad (2.19)$$

E' immediato ora vedere che il minimo di $D(p \parallel q)$, rispetto a $q(x)$, si ha per $q = p_T$. Infatti nell'ultimo passaggio della (2.19) il primo termine non dipende da $q(x)$, il secondo invece è sempre maggiore o uguale a 0 e risulta uguale a 0 se e solo se $q = p_T$.

Calcoliamo ora il termine $D(p \parallel p_T)$. Avremo:

$$\begin{aligned} D(p \parallel p_T) &= \int p(x) \log p(x)\, dx - \int p(x) \log p_T(x)\, dx = \\ &= -H(x) - \sum_{(u,v)\in\mathcal{E}} \int p(x) \log \frac{p_{uv}(x_u, x_v)}{p_u(x_u)\, p_v(x_v)} dx_u dx_v - \sum_{u\in\mathcal{V}} \int p(x) \log p_u(x_u)\, dx_u = \\ &= -H(x) - \sum_{(u,v)\in\mathcal{E}} I(x_u, x_v) + \sum_{u\in\mathcal{V}} H(x_u) = \\ &= I(x_1, \ldots, x_m) - \sum_{(u,v)\in\mathcal{E}} I(x_u, x_v) = I^T(x) \end{aligned}$$

$I^T(x)$ rappresenta la minima perdita possibile di informazione quando approssimiamo $p(x)$ con una $q(x)$ che fattorizza in T e risulta essere uguale a 0 se e solo se $p(x)$ fattorizza in T. Riprendiamo ora quanto detto sull'indipendenza condizionata. Abbiamo già definito gli insiemi di variabili x_A, x_B e x_C in modo che x_A e x_B sono indipendenti dato x_C se nell'albero T, C divide A da B (con A, B e C sottoinsiemi disgiunti di $\mathcal{V}$). Questo implica che $(x_1, ..., x_m)$ fattorizzano in T. Il teorema 2.1 ci asicura che:

$$I^T(x) = 0 \Rightarrow x_A \perp\!\!\!\perp x_B \,|x_C \quad (2.20)$$

Ora dimostreremo l'implicazione inversa ([20])

$$x_A \perp\!\!\!\perp x_B \,|x_C \Rightarrow I^T(x) = 0 \tag{2.21}$$

Partiamo dalla divergenza di KL tra p e p_T che rappresenta proprio la I^T. Osserviamo che, per l'ipotesi fatta nella (2.21), possiamo scrivere la p e la p_T come:

$$\begin{aligned} p(x) = p(x_A, x_B, x_C) = p_{A,B|C}(x_A, x_B \,|x_C)\, p_C(x_C) = \\ = p_{A|C}(x_A \,|x_C)\, p_{B|C}(x_B \,|x_C)\, p_C(x_C) \end{aligned} \tag{2.22}$$

$$\begin{aligned} p_T(x) = p_T(x_A, x_B, x_C) = p_{T_{A,B|C}}(x_A, x_B \,|x_C)\, p_{T_C}(x_C) = \\ = p_{T_{A|C}}(x_A \,|x_C)\, p_{T_{B|C}}(x_B \,|x_C)\, p_{T_C}(x_C) \end{aligned} \tag{2.23}$$

Avremo quindi che:

$$\begin{aligned} I^T(x) = D(p \parallel p_T) = \int p(x) \log \frac{p(x)}{p_T(x)} dx = \\ = \int p(x) \log p(x)\, dx - \int p(x) \log p_T(x)\, dx = \\ = -H(x) - \\ - \int p_{A|C}(x_A \,|x_C)\, p_{B|C}(x_B \,|x_C)\, p_C(x_C) \cdot \\ \cdot \log \Big(p_{T_{A|C}}(x_A \,|x_C)\, p_{T_{B|C}}(x_B \,|x_C)\, p_{T_C}(x_C) \Big) dx_A dx_B dx_C = \\ = -H(x) - \int p_{A|C}(x_A \,|x_C) \log p_{T_{A|C}}(x_A \,|x_C)\, dx_A - \\ - \int p_{B|C}(x_B \,|x_C) \log p_{T_{B|C}}(x_B \,|x_C)\, dx_B - \int p_C(x_C) \log p_{T_C}(x_C)\, dx_C \end{aligned} \tag{2.24}$$

Ora sfruttando un fatto dimostrato precedentemente, osserviamo che:

$$\int p_{A|C}(x_A \,|x_C) \log p_{T_{A|C}}(x_A \,|x_C)\, dx_A = \int p_{T_{A|C}}(x_A \,|x_C) \log p_{T_{A|C}}(x_A \,|x_C)\, dx_A$$

$$\int p_{B|C}(x_B \,|x_C) \log p_{T_{B|C}}(x_B \,|x_C)\, dx_B = \int p_{T_{B|C}}(x_B \,|x_C) \log p_{T_{B|C}}(x_B \,|x_C)\, dx_B$$

$$\int p_C(x_C) \log p_{T_C}(x_C)\, dx_B = \int p_{T_C}(x_C) \log p_{T_C}(x_C)\, dx_C$$

Sostituendo ora nella (2.24) avremo:

$$\begin{aligned} I^T(x) &= -H(x) + H(x_A|x_C) + H(x_B|x_C) + \\ &\quad + H(x_C) - H(x_A) + H(x_A) - H(x_B) + H(x_B) = \\ &\quad = I(x_A, x_B, x_C) - I(x_A, x_C) - I(x_B, x_C) \end{aligned} \quad (2.25)$$

Proviamo ora a scrivere in modo diverso la (2.25). Riscriviamo in forma esplicita i tre termini di informazione mutua e raccogliamoli in un unico integrale:

$$\begin{aligned} &I^T(x) = \\ &= \int p(x_A, x_B, x_C) \log \frac{p(x_A, x_B, x_C)\, p_A(x_A)\, p_B(x_B)\, p_C^2(x_C)}{p_A(x_A)\, p_B(x_B)\, p_C(x_C)\, p_{A,C}(x_A, x_C)\, p_{C,B}(x_C, x_B)} dx_A dx_B dx_C = \\ &= \int p(x_A, x_B, x_C) \log \frac{p_{A,B|C}(x_A, x_B|x_C)}{p_{A|C}(x_A|x_C)\, p_{B|C}(x_B|x_C)} dx_A dx_B dx_C = \\ &= \int p_{A|C,B|C}(x_A|x_C, x_B|x_C)\, p(x_C) \log \frac{p_{A|C,B|C}(x_A|x_C, x_B|x_C)}{p_{A|C}(x_A|x_C)\, p_{B|C}(x_B|x_C)} dx_A dx_B dx_C = \\ &= \int \left(\int p_{A|C,B|C}(x_A|x_C, x_B|x_C) \log \frac{p_{A|C,B|C}(x_A|x_C, x_B|x_C)}{p_{A|C}(x_A|x_C)\, p_{B|C}(x_B|x_C)} dx_A dx_B \right) p(x_C)\, dx_C \end{aligned}$$

dove abbiamo usato l'uguaglianza $p_{A,B|C}(x_A, x_B|x_C) = p_{A|C,B|C}(x_A|x_C, x_B|x_C)$. Possiamo quindi scrivere che:

$$I^T(x) = E_C\{I(x_A|x_C, x_B|x_C)\} \quad (2.26)$$

Se quindi $x_A \perp\!\!\!\perp x_B|x_C$ (vero per ipotesi), risulta immediatamente che $I(x_A|x_C, x_B|x_C) = 0$, di conseguenza anche $I^T(x) = 0$. Abbiamo così dimostrato l'implicazione (2.21). Possiamo quindi affermare che:

$$I^T(x) = 0 \Leftrightarrow x_A \perp\!\!\!\perp x_B|x_C$$

Poniamoci ora il problema di trovare l'albero T su cui modellare la $p(x)$ con la minore perdita di informazione. Questo problema si risolve facilmente minimizzando l'informazione mutua rispetto a T. Il primo termine di $I^T(x)$, cioè $I(x_1, \ldots, x_m)$, non dipende da T ed è sempre non negativo (come tutti i termini di informazione mutua). Il problema si riduce a massimizzare $\sum_{(u,v)\in\mathcal{E}} I(x_u, x_v)$. Dobbiamo quindi trovare lo *spanning tree* che massimizza i pesi (i termini $I(x_u, x_v)$) definiti sui suoi lati. Notiamo che questo algoritmo darà in uscita sempre uno spanning tree che potrebbe non

essere la soluzione ottima. Infatti come ipotesi abbiamo ammesso la possibilità che tra due vertici non ci sia nessun percorso che li colleghi (T potrebbe essere una foresta). Possiamo però modificare il nostro problema arricchendolo con le informazioni a priori che potremmo avere sulla foresta T. Possiamo definire la densità di probabilità della foresta T, $p(T)$. Definiamo poi una funzione $w(T) = \log p(T)$. Il nostro problema sarà ora quello di minimizzare rispetto a T la funzione $I^T(x) - w(T)$. Prenderemo in considerazione funzioni $w(T)$ della forma $w(T) = \sum_{(u,v)\in E} w^0_{uv} + f(\#(T))$, dove w^0_{uv} sono dei pesi fissati, $\#T$ è il numero di lati di T e f è una funzione concava. Ricordiamo che una funzione si dice *concava* su un insieme X se, per ogni $a, b \in X$ e per $\lambda \in (0,1)$, vale che: $f((1-\lambda)a + \lambda b) \geqslant (1-\lambda) f(a) + \lambda f(b)$. L'algoritmo che useremo è il cosiddetto greedy algoritm. Ne daremo ora la descrizione:

Input: pesi $\{w_{uv}; u, v \in V\}$, una funzione concava $f(t)$
Algoritmo:

1. Inizializzazione: $\mathcal{E}=\emptyset$, $t = 0$, $\mathcal{A}=\mathcal{V} \times \mathcal{V}$
2. Finchè $\mathcal{A} \neq \emptyset$
   (a) Trovare $w_{u_0 v_0} = \max_{(u,v)\in A} w_{uv}$
   (b) se $w_{u_0 v_0} + f(t+1) - f(t) > 0$
   $E \leftarrow E \cup (u_0, v_0)$, $\quad t \leftarrow t+1$
   $A \leftarrow \{e \in A, E \cup \{e\}\}$
   altrmenti $\mathcal{A} = \emptyset$

Output: $T(\mathcal{V}, \mathcal{E})$, foresta di peso massimo

---

Proposizione 2.1. *Se $J(T)$ è della forma $J(T) = \sum_{(u,v)\in E} w_{uv} + f(\#(T))$ dove $\{w_{uv}, u, v \in V\}$ è in insieme fissato di pesi e f è una funzione concava, allora il greedy algorithm descritto sopra fornisce il massimo globale di $J(T)$ su tutte le possibili foreste.*

Una dimostrazione di questo fatto può essere trovata in ([21]). Notiamo che i pesi di $w(T)$, se negativi e di modulo sufficientemente grande, possono penalizzare di molto l'aggiunta di nuovi lati.

Torniamo ora all'algoritmo TCA. Come abbiamo già detto assumiamo che il vettore delle sorgenti s e quello dei dati x siano legati da una trasformazione lineare invertibile tale che $x = As$. Sia ora $W = A^{-1}$, indichiamo con $\mathcal{D}^{W,T}$ l'insieme di tutte le densità di probabilità q_x tali che q_s fattorizza in T con $s = Wx$. Dato che la divergenza KL è invariante alle trasformazioni lineari invertibili possiamo estendere il Teorema 2.1 nel seguente modo:

Teorema 2.2. *Se $x \in \mathbb{R}^m$ ha densità di probabilità p_x, allora il minimo della divergenza KL tra p_x e una densità $q_x \in \mathcal{D}^{W,T}$ è uguale alla T-mutual information di $s = Wx$ e quindi:*

$$J(x, W, T) = \min_{q \in D^{W,T}} D(p \parallel q) = I^T(s) = \\ = I(s_1, \ldots, s_m) - \sum_{(u,v) \in \mathcal{E}} I(s_u, s_v)$$

Questo teorema ci fornisce la contrast function che sarà esattamente $J(x, W, T)$. Il nostro obiettivo sarà quello di minimizzare $J(x, W, T)$ rispetto a W e a T. Non conoscendo però la $p(x)$ dovremo cercare un modo per stimare la contrast function dai dati. Prima di occuparci di questo problema, affrontiamo il problema delle ambiguità e dell'algoritmo di minimizzazione.

2.5.4 Ambiguità dell'algoritmo TCA

Come abbiamo detto precedentemente, usando l'algoritmo ICA la demixing matrix W può essere stimata a meno di permutazioni e scalatura delle sue righe. Elencheremo ora alcune delle ambiguità dell'algoritmo TCA. E' importante notare però che le ambiguità riportate sono solo necessarie ma potrebbero essere non sufficienti. Attualmente infatti non esiste un lavoro che copra in maniera completa questo aspetto.

Permutazione delle componenti La demixing matrix W può essere premoltiplicata per una matrice di permutazione P senza cambiare il valore di $J(x, W, T)$ a patto di permutare nello stesso modo i vertici dell'albero T. Questo implica che inizialmente non dobbiamo considerare tutti i possibili alberi ma solo le classi di equivalenza definite dalla permutazione dei vertici.

Scalatura delle componenti Possiamo premoltiplicare W per una qualsiasi matrice diagonale reale e invertibile. La spiegazione è del tutto analoga a quella fatta per ICA e può essere riassunta dalla (2.6).

Combinazione lineare di un nodo foglia e del suo genitore In dato un albero T, se sostituiamo un nodo foglia s_c (figura 2.2) con una combinazione lineare $\lambda s_p + \mu s_c$ dove con s_p abbiamo indicato il nodo genitore, il valore della T-mutual information non cambia. Diamo ora una dimostrazione formale di quanto detto. Supponiamo che, a meno di permutazioni, s_p e s_c siano rispettivamente la prima e la seconda componente del vettore s. Indichiamo con T la trasformazione che agisce su s_c lasciando tutto il resto globalmente invariato. In formule:

$$T : \mathbb{R}^m \to \mathbb{R}^m$$

$$\begin{pmatrix} s_p \\ s_c \\ \vdots \end{pmatrix} \mapsto \begin{pmatrix} s_p \\ \mu s_c + \lambda s_p \\ \vdots \end{pmatrix}$$

con μ e λ reali e μ diverso da 0. T può quindi essere scritta nella forma:

$$T = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \lambda & \mu & 0 & \cdots & 0 \\ \vdots & & 1 & & 0 \\ 0 & & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix}$$

oppure, definendo $T' = \begin{pmatrix} 1 & 0 \\ \lambda & \mu \end{pmatrix}$, possiamo scrivere T in maniera più compatta come:

$$T = \begin{pmatrix} T' & 0 \\ 0 & I \end{pmatrix}$$

dove I indica la matrice identità. Indichiamo con s' il vettore risultante dalla trasformazione T: $s' = Ts$. Quello che ci proponiamo di dimostrare ora è che $I^T(s') = I^T(s)$. Nella dimostrazione useremo delle relazioni dimostrate nell'appen-

dice A.

$$I^T(s') = I(s') - \sum_{(u,v)\in\mathcal{E}} I(s'_u, s'_v) =$$

$$= \left(\sum_{u\in\mathcal{V}} H(s'_u) - H(s')\right) - \sum_{(u,v)\in\mathcal{E}} [H(s'_u) + H(s'_v) - H(s'_u, s'_v)] =$$

$$= \sum_{\substack{u\in\mathcal{V}\\ u\neq c}} H(s_u) + H(\lambda s_p + \mu s_c) - H(s') -$$

$$- \sum_{\substack{(u,v)\in\mathcal{E}\\ u\neq c, v\neq p}} I(s_u, s_v) - [H(\lambda s_p + \mu s_c) + H(s_p) - H(\lambda s_p + \mu s_c, s_p)] \tag{2.27}$$

Ricordiamo ora che, dalle proprietà dell'entropia differenziale, vale: $H(Ax) = H(x) + \log|\det A|$. Fruttando questa relazione e eliminando i termini opposti, possiamo riscrivere la (2.27) come:

$$I^T(s') = \sum_{\substack{u\in\mathcal{V}\\ u\neq c}} H(s_u) - H(s) - \log|\det T| -$$

$$- \sum_{\substack{(u,v)\in\mathcal{E}\\ u\neq c, v\neq p}} I(s_u, s_v) - H(s_p) + H(s_p, s_c) + \log|\det T'|$$

Osservando ora che $\log|\det T| = \log|\det T'|$ e aggiungendo e sottraendo $H(s_c)$ otteniamo:

$$I^T(s') =$$

$$= \sum_{\substack{u\in\mathcal{V}\\ u\neq c}} H(s_u) + H(s_c) - H(s) - \sum_{\substack{(u,v)\in\mathcal{E}\\ u\neq c, v\neq p}} I(s_u, s_v) - [H(s_p) + H(s_c) - H(s_p, s_c)] =$$

$$= \sum_{u\in\mathcal{V}} H(s_u) - H(s) - \sum_{(u,v)\in\mathcal{E}} I(s_u, s_v) = I(s) - \sum_{(u,v)\in\mathcal{E}} I(s_u, s_v) =$$

$$= I^T(s)$$

Questo conclude la dimostrazione.

Mentre le prime due ambiguità sono relativamente semplici da trattare, quest'ultima può dare risultati molto lontani dal modello originale. Osserviamo subito che questa ambiguità crea una

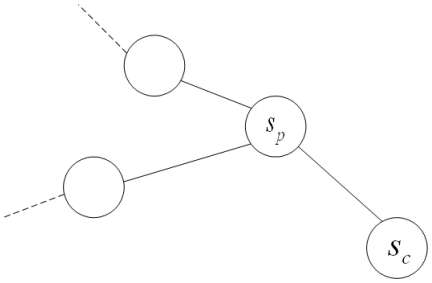


Figura 2.2: Un possibile esempio di nodo foglia s_c e di nodo genitore s_p

dipendenza lineare tra componenti che, in partenza non c'era. Una possibile soluzione può essere quella di imporre la decorrelazione tra nodo foglia e nodo parente.

2.5.5 Algoritmo di minimizzazione

Occupiamoci ora di formulare l'algoritmo di minimizzazione della contrast function.

Formulazione del problema di minimizzazione

Come abbiamo già detto, dobbiamo trovare il minimo della contrast function rispetto a una matrice W e ad un albero T. Come primo passo proviamo a ridurre lo spazio delle matrici sul quale effettuare la ricerca. Per far questo sfruttiamo a nostro favore una delle ambiguità del TCA, più precisamente quella ai fattori di scala. Indichiamo con Σ_x la matrice di covarianza dei dati e con Σ_s quella delle sorgenti stimate. Avremo che $\Sigma_s = W\Sigma_x W^t = \left(W\Sigma_x^{1/2}\right)\left(W\Sigma_x^{1/2}\right)^t$. Siccome non è possibile determinare l'esatta varianza delle sorgenti, tanto vale fissarla a 1, cioè: $(\Sigma_s)_{ii} = 1$ con $i \in \{1, \ldots, m\}$. Questa imposizione porta ad un vincolo su W, infatti avremo che: $(W\Sigma_x W^t)_{ii} = 1$ con $i \in \{1, \ldots, m\}$. In altre parole questo vincolo impone che le righe di $W\Sigma_x^{1/2}$ abbiano norma unitaria. Il problema di minimo che dobbiamo risolvere risulta quindi:

$$\min_{W,T} F(W,T) = \min_{W,T} J(x,W,T) - w(T)$$
$$\textbf{vincolo } (W\Sigma_x W^t)_{ii} = 1 \text{ con } i \in \{1, \ldots, m\}$$

Per semplicità, un questa sezione, indicheremo con W il prodotto

$W\Sigma_x^{1/2}$, questo non è un problema in quanto la matrice $\Sigma_x^{1/2}$ è nota.

Iniziamo osservando che in questo problema sono presenti sia una variabile continua W che una variabile discreta T. Dobbiamo quindi spezzare il problema: definiamo $G(W) = \min_T F(W,T)$. La minimizzazione rispetto a T può essere fatta efficientemente applicando il greedy algorithm già discusso. La funzione $G(W)$ è ora una funzione continua della variabile W. Dal vincolo che abbiamo imposto risulta che le righe di W devono avere norma unitaria, indichiamo quindi con $\mathcal{M}$ lo spazio delle matrici che soddisfano questa condizione. Scriviamo di seguito l'algoritmo utilizzato, ne daremo poi una spiegazione. Cominciamo dicendo

---

Input: Dati x
Algoritmo

1. Inizializzazione: W prodotta dall'algoritmo FastICA
2. Finchè $G(W) = \min_T F(W,T)$ è decrescente

   per $i = 2$ a m, per $j = 1$ a $i-1$, $W \leftarrow \underset{V \in L_{ij}(W)}{\arg\min} \left(\min_T F(V,T) \right)$

   dove $L_{ij}(W)$ è l'insieme delle matrici $V \in \mathcal{M}$ tale che:

   (a) $\forall k \notin \{i,j\}, V_k = W_k$
   (b) $Span(V_i, V_j) = Span(W_i, W_j)$
3. Calcolare $T = \arg\min F(W,T)$

Output: Demixing matrix W e l'albero T

---

che lo spazio $\mathcal{M}$ può essere generato partendo da una matrice $V \in \mathcal{M}$ qualsiasi e ruotando le sue righe (dato che la loro norma è fissa a 1). Partendo da questo si può pensare ad un algoritmo di minimizzazione che procede iterativamente: ad ogni iterazione viene scelta una coppia di indici (i,j), quindi tutte le righe di W vengono tenute fisse eccetto la i-esima e la j-esima. Queste due righe vengono ruotate di due angoli θ_i e θ_j. Abbiamo così ridotto il nostro problema: ad ogni iterazione non cerchiamo il minimo rispetto ad una matrice ma ripetto a due variabili monodimensionali (i due angoli). Ci siamo così ricondotti ad un problema di minimo in due dimensioni. Questo problema bidimensionale può essere risolto efficientemente tramite l'algoritmo del gradiente coniugato.

2.5.6 Stima della contrast function: metodo Kernel Density Estimation (KDE)

In questa sezione descriveremo due diverse contrast function, la prima sfrutta un metodo diretto (stima dell'entropia) mentre la seconda è basata sul recente lavoro di Bach e Jordan e prende il nome di *Kernel Generalized Variance*. Ricordiamo qui l'espressione analitica della contrast function, che nel nostro caso è la T-mutual information, cioè: $J(x, W, T) = I^T(s) = I(s_1, \ldots, s_m) - \sum_{(u,v)\in E} I(s_u, s_v)$. Noi possediamo solo i dati osservati e da questi dobbiamo stimare i termini di informazione mutua. Cominciamo col descrivere la prima contrast function.

Stima dell'entropia

Il primo tipo di contrast function è basato sulla stima dell'entropia. I termini di informazione mutua della $J(x, W, T)$ possono infatti essere riscritti come:

- $I(s_1, \ldots, s_m) = \sum_{u\in V} H(s_u) - H(s)$
- $I(s_u, s_v) = H(s_u) + H(s_v) - H(s_u, s_v)$

Ora, ricordando che $s = Wx$, per le note proprietà dell'entropia avremo che $H(s) = H(Wx) = H(x) + \log|\det W|$. Il termine $H(x)$ che non dipende ne dall'albero T ne dalla matrice W può essere tralasciato. Dovremo quindi minimizzare la seguente funzione:

$$J(x, W, T) = \sum_{u\in V} H(s_u) - \sum_{(u,v)\in E} (H(s_u) + H(s_v) - H(s_u, s_v)) - \log|\det W|$$

Dobbiamo quindi stimare i termini di entropia marginale $H(s_u)$ e quelli di entropia congiunta $H(s_u, s_v)$. Per far questo andremo prima a stimare la densita di probabilità marginale $\hat{f}_{s_u}(s_u)$ e quella congiunta $\hat{f}_{s_u,s_v}(s_u, s_v)$ poi, tramite integrazione numerica otterremo le relative entropie differenziali, più precisamente:

- $\hat{H}(s_u) = -\int \hat{f}_{s_u}(s_u) \log \hat{f}_{s_u}(s_u)\, ds_u$
- $\hat{H}(s_u, s_v) = -\int \hat{f}_{s_u,s_v}(s_u, s_v) \log \hat{f}_{s_u,s_v}(s_u, s_v)\, ds_u ds_v$

La tecnica che useremo per stimare le densità di probabilità e quella *KDE* cioè *Kernel Density Estimation* ([22]).

Kernel Density Estimation Dato un insieme di N campioni $x_i \in \mathbb{R}^d$ con $i = 1, \ldots, N$, la tecnica KDE ci permette di stimare la sua densità di probabilità $f(x)$.

Definiamo *nucleo* una funzione K tale che:

- $K : \mathbb{R}^d \to \mathbb{R}$ sia non negativa
- $\int K(x)\, dx = 1$

La stima della densità di probabilità sarà data da:

$$\hat{f}(x) = \frac{1}{Nh^d} \sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right) \tag{2.28}$$

Noi useremo il nucleo Gaussiano $K(x) = \frac{1}{(2\pi)^{d/2}} e^{-\|x\|^2/2}$.

Da un punto di vista computazionale, l'applicazione diretta della (2.28) ha una bassa efficienza e un alto costo. Se, infatti, il numero di campioni N è molto elevato, il calcolo diretto della (2.28) diventa proibitivo. Descriveremo ora un metodo indiretto ma molto più veloce più basato sulla trasformata di Fourier.

Cominciamo descrivendo il caso monodimensionale, l'estensione a due variabili risulterà immediata.

Partiamo facendo la trasformata di Fourier della (2.28):

$$\begin{aligned}
\hat{\mathcal{F}}(\omega) &= \int \hat{f}(x)\, e^{-i\omega x} dx = \\
&= \int \frac{1}{Nh} \sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right) e^{-i\omega x} dx = \\
&= \frac{1}{Nh} \sum_{i=1}^{N} \int K\left(\frac{x - x_i}{h}\right) e^{-i\omega x} dx \overset{x - x_i = a}{=} \\
&= \frac{1}{Nh} \sum_{i=1}^{N} \int K\left(\frac{a}{h}\right) e^{-i\omega a} e^{-i\omega x_i} da = \\
&= \frac{1}{Nh} \mathcal{K}\left(\frac{\omega}{h}\right) \sum_{i=1}^{N} e^{-i\omega x_i}
\end{aligned} \tag{2.29}$$

Nell'ultimo passaggio della (2.29) abbiamo due termini: il primo, $\frac{1}{h}\mathcal{K}\left(\frac{\omega}{h}\right)$, rappresenta la trasformata di Fourier della funzione nucleo (a meno della costante $1/h$), mentre il secondo, $\frac{1}{N}\sum_{i=1}^{N} e^{-i\omega x_i}$, può essere approssimato con ottimi risultati dalla trasformata discreta di Fourier dell'istogramma dei dati. In particolare, indicando con $h\left(n\right)$ l'istogramma dei dati, avremo che:

$$\frac{1}{N}\sum_{n=1}^{N} h\left(n\right) e^{-i\frac{\omega n}{N}} \simeq \frac{1}{N}\sum_{i=1}^{N} e^{-i\omega x_i} \tag{2.30}$$

Ovviamente l'accuratezza della stima dipende dal numero di punti su cui si calcola la trasformata discreta di Fourier. Questo parametro può essere scelto a piacere.

Per ottenere infine la densità di probabilità voluta basta prendere l'antitrasformata di Fourier della (2.29), cioè:

$$\hat{f}\left(x\right) = \mathfrak{F}^{-1}\left[\hat{\mathcal{F}}\left(\omega\right)\right]$$

L'estensione al caso bidimensionale è immediata: basta ripercorrere esattamente gli stessi passaggi usando però trasformate e antitrasformate di Fourier bidimensionali.

2.5.7 Stima della contrast function: metodo Kernel Generalized Variance (KGV)

Il secondo tipo di contrast function che analizzeremo si basa sulla stima dell'informazione mutua tramite un procedimento detto *KGV* ovvero *Kernel Generalized Variance* ([23]). Iniziamo spiegando l'idea di base: sia $x_G \in \mathbb{R}^m$ un vettore aleatorio Gaussiano e sia Σ_G la sua matrice di covarianza. Si può dimostrare ([24]) che l'informazione mutua risulta essere uguale a:

$$I\left(x_G\right) = -\frac{1}{2}\log\left(\frac{\det\Sigma_G}{\left(\Sigma_G\right)_{11}\cdots\left(\Sigma_G\right)_{mm}}\right) \tag{2.31}$$

dove il rapporto $\frac{\det\Sigma_G}{\left(\Sigma_G\right)_{11}\cdots\left(\Sigma_G\right)_{mm}}$ è comunemente detto *varianza generalizzata*. Se quindi il nostro vettore fosse Gaussiano potremmo esprimere la nostra contrast function in funzione della sola

matrice di covarianza Σ_G. Basta infatti scrivere:

$$J(x_G, W, T) = I^T(x_G) = \\ = -\frac{1}{2}\log\left(\frac{\det\Sigma_G}{(\Sigma_G)_{11}\cdots(\Sigma_G)_{mm}}\right) + \frac{1}{2}\log\left(\frac{\det\Sigma_G^{uv}}{(\Sigma_G^{uv})_{11}\cdots(\Sigma_G^{uv})_{mm}}\right) \tag{2.32}$$

dovela matrice Σ_G^{uv} rappresenta la matrice di covarianza del vettore $\begin{pmatrix} x_u & x_v \end{pmatrix}^t$. Se però x non è Gaussiano l'applicazione diretta di questo metodo non porta a buoni risultati.

Possiamo però pensare di mappare ciascuna componente di x in uno spazio di funzioni $\mathcal{F}$ e trattare le variabili proiettate come se fossero gaussiane in $\mathcal{F}$. Più precisamente indicando con ϕ una mappa da $\mathbb{R}$ in $\mathcal{F}$ tale che $\phi(x) = (\phi(x_1), \ldots, \phi(x_m))^t$, e definendo una "matrice" di covarianza Σ per il vettore $\phi(x)$ possiamo applicare la (2.32).

Dopo aver dato questa idea generale diamo una trattazione matematica più precisa. Questa trattazione sarà abbastanza lunga e non è strettamente legata all'algoritmo TCA, ma è stata inserita in questa tesi per ottenere un lavoro il più completo possibile. Cominceremo definendo il concetto di *Reproducing Kernel Hilbert Space* (RKHS). Passeremo poi a introdurre l'algoritmo CCA (*Canonical Correlation Analysis*) e KCCA (*Kernel CCA*), questo ci servirà per capire gli aspetti cruciali della tecnica KGV. Analizzeremo infine i limiti di questa tecnica, preannunciando fin d'ora che una dimostrazione delle sue proprietà è stata trovata solo nel caso di due variabili aleatorie.

Reproducing Kernel Hilbert Space

Cominciamo definendo il concetto di funzione *kernel* ([25]).

Definizione

La funzione $k(x, y)$ con $x, y \in E$ con $E \subseteq \mathbb{R}^p$ è detta *reproducing kernel* (RK) di F, dove F è una classe di funzioni se:

- $\forall y \in E$
  $k(x, y)$ come funzione di x appartiene a F

- *Reproducing property*
  $\forall y \in E, \forall f \in F$
  $f(y) = \langle f(x), k(x,y) \rangle_x$

Per ogni funzione $k(x,y)$ esiste unico uno spazio di Hilbert (cioè una classe di funzioni F e un prodotto scalare) che ammette $k(x,y)$ come RK. Questo spazio di Hilbert è detto *reproducing kernel Hilbert space*, abbreviato con RKHS. La classe di funzioni F è generata da combinazioni lineari del tipo

$$\sum_i a_i k(x, y_i) \tag{2.33}$$

In realtà dato un generico kernel, le funzioni generate da queste combinazioni lineari non formano uno spazio completo. Si può dimostrare però che successioni di Cauchy di queste funzioni convergono a una funzione limite la cui aggiunta alla classe forma uno spazio di Hilbert.

Ci sono poi dei particolari kernel, detti kernel universali, che generano spazi completi. Un esempio di questi è il kernel Gaussiano

$$k(x,y) = G_\sigma(x-y) = \exp\left(-\frac{1}{2\sigma^2}\|x-y\|^2\right) \tag{2.34}$$

Si può dimostrare ([26]) che l'RKHS generato dal kernel gaussiano si ottiene dalla convoluzione tra $G_{\sigma/\sqrt{2}}(x) = \exp\left(-\frac{1}{\sigma^2}\|x\|^2\right)$ e una $f \in L_2$.

Torneremo su questo argomento più avanti. Introduciamo ora l'algoritmo CCA.

Canonical Correlation Analysis: *CCA*

Dati 2 vettori aleatori x,y (supponiamoli per ora a media nulla) di dimensioni p_1 e p_2, la tecnica CCA ([23]) ci permette di trovare 2 vettori w_x e w_y tali che le due variabili aleatorie $\langle w_x, x\rangle$ e $\langle w_y, y\rangle$ siano massimamente correlate.

Definiamo infatti:

$$\begin{aligned}\rho(x,y) &= \max_{w_x,w_y} \operatorname{corr}\left\{w_x^t x, w_y^t y\right\} \\ &= \max_{w_x,w_y} \frac{\operatorname{cov}\left\{w_x^t x, w_y^t y\right\}}{\operatorname{var}\left\{w_x^t x\right\}^{1/2} \operatorname{var}\left\{w_y^t y\right\}^{1/2}}\end{aligned} \tag{2.35}$$

dove

$$\begin{aligned}\operatorname{cov}\left\{w_x^t x, w_y^t y\right\} &= E\left\{\left(w_x^t x\right)\left(w_y^t y\right)\right\} \\ &= w_x^t E\left\{xy^t\right\} w_y = w_x^t C_{xy} w_y\end{aligned} \tag{2.36}$$

possiamo quindi scrivere

$$\rho\left(x,y\right) = \max_{w_x, w_y} \frac{w_x^t C_{xy} w_y}{\left(w_x^t C_{xx} w_x\right)^{1/2}\left(w_y^t C_{yy} w_y\right)^{1/2}} \tag{2.37}$$

Si vede subito che ρ è invariante a fattori di scala su w_x e w_y. Possiamo quindi imporre che:

$$w_x^t C_{xx} w_x = 1 \tag{2.38}$$

$$w_y^t C_{yy} w_y = 1 \tag{2.39}$$

e massimizzare il numeratore della (2.37) usando i moltiplicatori di Lagrange.

La funzione Lagrangiana sarà:

$$\begin{aligned}&L\left(\lambda_x, \lambda_y, w_x, w_y\right) = \\ &\qquad = w_x^t C_{xy} w_y - \frac{\lambda_x}{2}\left(w_x^t C_{xx} w_x - 1\right) - \frac{\lambda_y}{2}\left(w_y^t C_{yy} w_y - 1\right)\end{aligned} \tag{2.40}$$

Facciamo ora le derivate della (2.40) rispetto alle componenti di w_x e w_y

$$\frac{\partial L}{\partial w_x} = C_{xy} w_y - \lambda_x C_{xx} w_x = 0 \tag{2.41}$$

$$\frac{\partial L}{\partial w_y} = C_{yx} w_x - \lambda_y C_{yy} w_y = 0 \tag{2.42}$$

moltiplichiamo ora la (1.21) per w_y^t e la (1.20) per w_x^t, poi ne facciamo la differenza

$$\begin{aligned}0 &= w_x^t C_{xy} w_y - \lambda_x \underbrace{w_x^t C_{xx} w_x}_{1} - w_y^t C_{yx} w_x + \lambda_y \underbrace{w_y^t C_{yy} w_y}_{1} = \\ &= w_x^t C_{xy} w_y - \left(w_x^t C_{xy} w_y\right) - \lambda_x + \lambda_y\end{aligned} \tag{2.43}$$

da qui segue che $\lambda_x = \lambda_y$. Otteniamo quindi il sistema

$$\begin{cases} C_{xy}w_y - \lambda C_{xx}w_x = 0 \\ C_{yx}w_x - \lambda C_{yy}w_y = 0 \end{cases} \tag{2.44}$$

che scritto in forma matriciale risulta essere:

$$\begin{pmatrix} 0 & C_{xy} \\ C_{yx} & 0 \end{pmatrix} \begin{pmatrix} w_x \\ w_y \end{pmatrix} = \lambda \begin{pmatrix} C_{xx} & 0 \\ 0 & C_{yy} \end{pmatrix} \begin{pmatrix} w_x \\ w_y \end{pmatrix} \tag{2.45}$$

Abbiamo ottenuto un problema generalizzato agli autovalori. Il problema (2.45) ammette come soluzioni $\{\lambda_1, -\lambda_1, ..., \lambda_p, -\lambda_p, 0, ..., 0\}$ con $p = \min\{p_1, p_2\}$. E' utile riscrivere il problema (2.45) nella forma

$$\begin{pmatrix} C_{xx} & C_{xy} \\ C_{yx} & C_{yy} \end{pmatrix} \begin{pmatrix} w_w \\ w_y \end{pmatrix} = (1+\lambda) \begin{pmatrix} C_{xx} & 0 \\ 0 & C_{yy} \end{pmatrix} \begin{pmatrix} w_w \\ w_y \end{pmatrix} \tag{2.46}$$

con soluzioni $\beta = (1+\lambda) \in \{1+\lambda_1, 1-\lambda_1, ..., 1+\lambda_1, 1-\lambda_1, 1, ...1\}$.

Ci sono molti modi per generalizzare la tecnica CCA ad un insieme di m vettori aleatori. Quello a noi più utile è la generalizzazione del problema (2.45) fatta nel seguente modo:

$$\begin{pmatrix} C_{11} & C_{12} & \cdots & C_{1m} \\ C_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ C_{m1} & C_{m2} & \cdots & C_{mm} \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{pmatrix} = \beta \begin{pmatrix} C_{11} & 0 & \cdots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & C_{mm} \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{pmatrix} \tag{2.47}$$

Vedremo più avanti come l'intero spettro di questo problema generalizzato agli autovalori abbia un forte legame con l'informazione mutua.

Kernel Canonical Correlation Analysis: *Kernel CCA*

Vediamo ora come sia possibile estendere l'algoritmo CCA in un RKHS ([23], [28]). Siano x e y due variabili aleatorie e siano f e g due funzioni appartenenti a due spazi di Hilbert (ovviamente possono anche essere lo stesso spazio) $\mathcal{F}$ e $\mathcal{G}$ rispettivamente.

Per ogni $f \in \mathcal{F}$ e per ogni $g \in \mathcal{G}$, definiamo l'operatore di covarianza $C_{xy} : \mathcal{G} \to \mathcal{F}$ come:

$$\operatorname{cov}\{f, g\} = \langle f, C_{xy} g\rangle_{\mathcal{F}} = E\left\{\left(f\left(x\right) - E\left\{f\left(x\right)\right\}\right)\left(g\left(x\right) - E\left\{g\left(x\right)\right\}\right)\right\} \tag{2.48}$$

Definiamo una mappa ϕ tale che:

$$\begin{aligned} \phi : & \chi \to H \\ & x \mapsto f \end{aligned}$$

con $\chi \subseteq \mathbb{R}^p$. H sarà invece un RKHS con kernel $k\left(x, y\right)$. La mappa ϕ sarà scelta in modo che

$$k(x, y) = \langle \phi(x), \phi(y) \rangle \tag{2.49}$$

In altre parole, basta fissare

$$\phi(x) = k(\cdot, x) \tag{2.50}$$

infatti, segue immediatamente dalla reproducing property che:

$$\langle \phi(x), \phi(y) \rangle = \langle k(\cdot, x), k(\cdot, y) \rangle = k(x, y) \tag{2.51}$$

Definiamo ora l'algoritmo CCA nello spazio RKHS.

Dati due kernel k_1 e k_2 con i rispettivi RKHS H_1 e H_2, definiamo due mappe $\phi_1 : \chi \to H_1$ e $\phi_2 : \chi \to H_2$ tali che:

$$\phi_1 = k_1(\cdot, x) \qquad\qquad \phi_2 = k_2(\cdot, x) \tag{2.52}$$

Dati poi due vettori aleatori x_1 e x_2, la tecnica kernel CCA (KCCA) ci permette di trovare due funzioni $f_1 \in H_1$ e $f_2 \in H_2$ tali che la correlazione tra $\langle \phi_1(x_1), f_1 \rangle$ e $\langle \phi_2(x_2), f_2 \rangle$ sia massima. Definiamo quindi ρ_F come:

$$\begin{aligned} \rho_F &= \max_{(f_1, f_2 \in H_1 \times H_2)} corr\left\{\left\langle \phi_1\left(x_1\right), f_1\right\rangle, \left\langle \phi_1\left(x_1\right), f_1\right\rangle\right\} = \\ &= \max_{(f_1, f_2 \in H_1 \times H_2)} corr\left\{f_1(x_1), f_2(x_2)\right\} \end{aligned} \tag{2.53}$$

l'ultimo passaggio segue immediatamente dalla reproducing property. Per semplicità, d'ora in poi useremo un solo kernel e quindi un solo spazio di funzioni.

Stima dello spettro del KCCA

Dato che a noi interessa lo spettro del KCCA per la sua relazione con l'informazione mutua, dobbiamo trovare un modo per stimarlo dai dati. In questa sezione cercheremo quindi di ricondurre l'equazione (2.53) ad un problema generalizzato agli autovalori simile a quello (2.45).

Indichiamo con $\left\{x_1^1, ..., x_1^N\right\}$ e $\left\{x_2^1, ..., x_2^N\right\}$ gli insiemi delle osservazioni. Supponiamo anche che queste siano a media nulla in H, cioè $\left(\sum_{i=1}^{N} \phi\left(x_1^i\right)\right)(y) = \left(\sum_{j=1}^{N} \phi\left(x_2^j\right)\right)(y) = 0$ (consultare Appendice B.1 per vedere come rimuovere questa ipotesi). Avremo che:

$$corr\left\{\left\langle \phi\left(x_1\right), f_1\right\rangle, \left\langle \phi\left(x_2\right), f_2\right\rangle\right\} = \operatorname{cov}\left\{\left\langle \phi\left(x_1\right), f_1\right\rangle, \left\langle \phi\left(x_2\right), f_2\right\rangle\right\} = \\ = \frac{1}{N}\sum_{1=1}^{N}\left\langle \phi\left(x_1^i\right), f_1\right\rangle\left\langle \phi\left(x_2^i\right), f_2\right\rangle \tag{2.54}$$

Indichiamo ora con S_1 e S_2 gli spazi generati dalle osservazioni

$$S_1 = Span\left(\left\{\phi\left(x_1^k\right)\right\}_{k=1}^{N}\right) \tag{2.55}$$

$$S_2 = Span\left(\left\{\phi\left(x_2^k\right)\right\}_{k=1}^{N}\right) \tag{2.56}$$

Definiamo ora $S_1^\perp$ e $S_2^\perp$ in modo che $H = S_1 \oplus S_1^\perp$ e anche $H = S_2 \oplus S_2^\perp$. Possiamo sempre scrivere f_1 e f_2 come

$$f_1 = \sum_{i=1}^{N} \alpha_1^i \phi\left(x_1^i\right) + f_1^\perp \qquad \text{con } f_1^\perp \in S_1^\perp \tag{2.57}$$

$$f_2 = \sum_{i=1}^{N} \alpha_2^i \phi\left(x_2^i\right) + f_2^\perp \qquad \text{con } f_2^\perp \in S_2^\perp \tag{2.58}$$

Sostituiamo le f_1 e f_2 così trovate nella (2.53).

$$\begin{aligned}
&\operatorname{cov}\left\{\left\langle \phi\left(x_1\right), f_1\right\rangle, \left\langle \phi\left(x_2\right), f_2\right\rangle\right\} = \\
&= \frac{1}{N}\sum_{i=1}^{N}\left\langle \phi\left(x_1^i\right), \sum_{j=1}^{N}\alpha_1^j\phi\left(x_1^j\right) + f_1^{\perp}\right\rangle\left\langle \phi\left(x_2^i\right), \sum_{k=1}^{N}\alpha_2^k\phi\left(x_2^k\right) + f_2^{\perp}\right\rangle = \\
&= \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}\alpha_1^j\alpha_2^k k\left(x_1^i, x_1^j\right) k\left(x_2^i, x_2^k\right) = \\
&= \frac{1}{N}\alpha_1^t K_1 K_2 \alpha_2 \qquad (2.59)
\end{aligned}$$

Con lo stesso calcolo avremo che:

$$\operatorname{var}\left\{\left\langle \phi\left(x_1\right), f_1\right\rangle\right\} = \frac{1}{N}\alpha_1^t K_1 K_1 \alpha_1 \qquad (2.60)$$

$$\operatorname{var}\left\{\left\langle \phi\left(x_2\right), f_2\right\rangle\right\} = \frac{1}{N}\alpha_2^t K_2 K_2 \alpha_2 \qquad (2.61)$$

Possiamo infine scrivere che

$$\hat{\rho}_F\left(K_1, K_2\right) = \max_{\alpha_1,\alpha_2\in\mathbb{R}^N} \frac{\alpha_1^t K_1 K_2 \alpha_2}{\left(\alpha_1^t K_1^2 \alpha_1\right)^{1/2}\left(\alpha_2^t K_2^2 \alpha_2\right)^{1/2}} \qquad (2.62)$$

questo è lo stesso problema risolto precedentemente per il CCA. Avremo quindi il problema agli autovalori del tipo

$$\begin{pmatrix} 0 & K_1K_2 \\ K_2K_1 & 0 \end{pmatrix}\begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} = \rho\begin{pmatrix} K_1^2 & 0 \\ 0 & K_2^2 \end{pmatrix}\begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} \qquad (2.63)$$

E' possibile rimuovere l'ipotesi di dati centranti trasformando in modo opportuno K. La dimostrazione di questo si trova nell'appendice B.1. Da qui in poi, ovunque useremo una matrice di Gram, la penseremo centrata.

Purtroppo però la stima trovata in (2.62) non è una buona stima. Per rendercene conto poniamo $v_1^t = \alpha_1^t K_1$ e $v_2^t = \alpha_2^t K_2$. Indichiamo con V_1 e V_2 i sottospazi generati dalle colonne di K_1 e K_2. Riscriviamo ora $\hat{\rho}_F$ con le sostituzioni fatte. Avremo:

$$\hat{\rho}_F = \max_{v_1\in V_1, v_2\in V_2} \frac{v_1^t v_2}{\left(v_1^t v_1\right)^{1/2}\left(v_2^t v_2\right)^{1/2}} = \max_{v_1\in V_1, v_2\in V_2} \cos\left(v_1, v_2\right) \qquad (2.64)$$

Le matrici K_1 e K_2 sono invertibili, quindi le colonne sono linearmente indipendenti. I sottospazi V_1 e V_2 da esse generati rappresenteranno tutto $\mathbb{R}^N$, quindi $\max_{v_1 \in V_1, v_2 \in V_2} \cos(v_1, v_2) \equiv 1$ indipendentemente dai dati.

Definiamo un nuovo stimatore introducendo un fattore di regolarizzazione.

$$\hat{\rho}_F^{\varepsilon} = \max_{f_1, f_2 \in H} \frac{\operatorname{cov}\{f_1(x_1), f_2(x_2)\}}{\left(\operatorname{var}\{f_1(x_1)\} + \varepsilon \|f_1\|^2\right)^{1/2} \left(\operatorname{var}\{f_2(x_2)\} + \varepsilon \|f_2\|^2\right)^{1/2}} \tag{2.65}$$

Questo stimatore, a differenza del (2.62) gode della proprietà di consistenza ([27]). Si può dimostrare ([27]) che, per garantire la convergenza nel caso generale, ε deve essere dell'ordine di $N^{-1/3}$. Scriviamolo in funzione delle matrici K_1 e K_2. Dato che $\|f_1\|^2 = \langle f_1, f_1 \rangle = \alpha_1^t K_1 \alpha_1$, avremo che:

$$\begin{aligned} \operatorname{var}\{f_1(x_1)\} + \varepsilon \|f_1\|^2 &= \frac{1}{N} \alpha_1^t K_1^2 \alpha_1 + \varepsilon \alpha_1^t K_1 \alpha_1 = \\ &= \frac{1}{N} \alpha_1^t \left(K_1^2 + N\varepsilon K_1\right) \alpha_1 \end{aligned} \tag{2.66}$$

Per semplicità di calcolo, facciamo la seguente approssimazione:

$$\left(K_1^2 + N\varepsilon K_1\right) = \left(K_1^2 + N\varepsilon K_1 + \frac{N^2\varepsilon^2}{2} I^2 - \underbrace{\frac{N^2\varepsilon^2}{2} I^2}_{\substack{\text{lo trascuro} \\ \varepsilon \ll 1}} \right) \simeq \left(K_1 + \frac{N\varepsilon}{2} I\right)^2 \tag{2.67}$$

Quindi lo stimatore sarà del tipo:

$$\hat{\rho}_F^{\varepsilon} = \max_{\alpha_1, \alpha_2 \in \mathbb{R}^N} \frac{\alpha_1^t K_1 K_2 \alpha_2}{\left(\alpha_1^t \left(K_1 + \frac{N\varepsilon}{2} I\right)^2 \alpha_1\right)^{1/2} \left(\alpha_2^t \left(K_2 + \frac{N\varepsilon}{2} I\right)^2 \alpha_2\right)^{1/2}} \tag{2.68}$$

Il problema agli autovalori corrispondente sara:

$$\begin{pmatrix} 0 & K_1 K_2 \\ K_2 K_1 & 0 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} = \rho \begin{pmatrix} \left(K_1 + \frac{N\varepsilon}{2} I\right)^2 & 0 \\ 0 & \left(K_2 + \frac{N\varepsilon}{2} I\right)^2 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} \tag{2.69}$$

La generalizzazione a più di due variabili, segue esattamente da quanto già fatto nella tecnica CCA. Supponiamo di avere m vettori aleatori, il problema da risolvere sarà:

$$\begin{pmatrix} \left(K_1 + \frac{N\varepsilon}{2} I\right)^2 & K_1 K_2 & \cdots & K_1 K_m \\ K_2 K_1 & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ K_m K_1 & \cdots & \cdots & \left(K_m + \frac{N\varepsilon}{2} I\right)^2 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_m \end{pmatrix} = \\ = \lambda \begin{pmatrix} \left(K_1 + \frac{N\varepsilon}{2} I\right)^2 & 0 & \cdots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & \left(K_m + \frac{N\varepsilon}{2} I\right)^2 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_m \end{pmatrix} \tag{2.70}$$

Come possiamo notare il problema (2.47) trovato per il CCA è formalmente identico al problema (2.70) appena formulato per il KCCA.

KGV (Kernel Generalized Variance) e informazione mutua

Possiamo cominciare ora a vedere come sfruttare i risultati ottenuti per la stima della contrast function KGV ([23], [28]). Quello che ci interessa è trovare una buona stima dell'informazione mutua tra un insieme di vettori aleatori o, più in particolare, tra un insieme di variabili aleatorie. Vedremo che questa stima può essere ottenuta sfruttando gli autovalori del problema (2.70).

Come inizio, dimostreremo che se i vettori sono gaussiani, gli autovalori del problema (2.45) forniscono il valore esatto dell'informazione mutua.

Siano x_1 e x_2 due vettori aleatori gaussiani di dimensioni p_1 e p_2 rispettivamente. Indichiamo con $C = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix}$ la matrice ottenuta da:

$$C = \left\{ \begin{pmatrix} x_1 & x_1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \right\} \tag{2.71}$$

L'informazione mutua tra x_1 e x_2 è definita come

$$I(x_1, x_2) = \int p(x_1, x_2) \log \left(\frac{p(x_1, x_2)}{p_{x_1}(x_1)\, p_{x_2}(x_2)} \right) dx_1 dx_2 \quad (2.72)$$

Per vettori gaussiani si può dimostrare che la (2.72) vale

$$I(x_1, x_2) = -\frac{1}{2} \log \left(\frac{\det C}{\det C_{11} \det C_{22}} \right) \quad (2.73)$$

Facendo opportuni passaggi matematici (vedere appendice B.2) arriviamo a scrivere che

$$I(x_1, x_2) = -\frac{1}{2} \log \left(\prod_{i=1}^{p} (1 - \rho_i^2) \right) = -\frac{1}{2} \sum_{i=1}^{p} \log (1 - \rho_i^2) \quad (2.74)$$

dove i ρ_i sono gli autovalori del problema

$$\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = (1 + \rho) \begin{pmatrix} C_{11} & 0 \\ 0 & C_{22} \end{pmatrix} \quad (2.75)$$

uguale a quello risolto per il CCA. Vale anche la generalizzazione a un insieme di m vettori gaussiani. Infatti l'informazione mutua tra m vettori gaussiani è data da:

$$I(x_1, ..., x_m) = -\frac{1}{2} \log \left(\frac{\det C}{\det C_{11} \cdots \det C_{mm}} \right) = -\frac{1}{2} \sum_{i=1}^{p} \log(1 - \rho_i)^2 \quad (2.76)$$

con ρ_i soluzioni del problema

$$\begin{pmatrix} C_{11} & C_{12} & \cdots & C_{1m} \\ C_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ C_{m1} & \cdots & \cdots & C_{mm} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_m \end{pmatrix} = (1 + \rho) \begin{pmatrix} C_{11} & 0 & \cdots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & C_{mm} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_m \end{pmatrix} \quad (2.77)$$

che possiamo scrivere in maniera compatta come: $Cv = (1 + \rho)Dv$, indichiamo poi il rapporto $\frac{\det C}{\det D}$ con il nome di *varianza generalizzata*. Notiamo subito che anche questo problema generalizzato è assolutamente identico al problema (2.47) trovato per il CCA. Il forte legame appena trovato tra l'informazione mutua di vettori gaussiani e gli autovalori del problema (2.77), ci spinge a chiederci se sia possibile estendere questi risultati al caso generale (non solo gaussiano). Diamo ora un'idea di quello che vogliamo fare:

- Proiettiamo gli m vettori aleatori $x_1, ..., x_m$ in un opportuno RKHS $\mathcal{F}$ tramite la mappa ϕ definita in precedenza.
- Prendiamo m vettori aleatori gaussiani in $\mathcal{F}$ con la stessa struttura di covarianza di $\phi(x_1), ..., \phi(x_m)$.
- Calcoliamo l'informazione mutua I^G tra i vettori aleatori gaussiani così ottenuti risolvendo il problema agli autovalori (2.77) e sfruttando poi la relazione (2.76). Risolvere il problema (2.77) in un RKHS significa stimare dai dati l'intero spettro. Abbiamo visto come questo sia possibile: basta infatti risolvere il problema (2.70).
- Useremo I^G come stima dell'informazione mutua I tra i vettori aleatori $x_1, ..., x_m$.

Non esiste una dimostrazione generale della validità di questo procedimento. Esiste una prova nel caso di due variabili aleatorie. Di seguito daremo la dimostrazione in questo caso particolare e prenderemo spunto da essa per fare alcune considerazioni generali sul procedimento adottato.

Dimostrazione nel caso di due variabili aleatorie

Siano x e y due variabili aleatorie e siano X e Y due sottoinsiemi chiusi di $\mathbb{R}$ tali che $(x, y) \in X \times Y$ ([23], [28]). Indichiamo con p_{xy} la densità di probabilità congiunta di x e y. Introduciamo l'approssimazione discreta dell'informazione mutua, considerando una griglia di dimensioni $l_x \times l_y$ su $X \times Y$ con spaziatura Δx e Δy. Usiamo gli indici i e j per indicare i punti su questa griglia: $(q_i, r_j) \in X \times Y$. Definiamo poi due vettori $q \in \mathbb{R}^{lx}$ e $r \in \mathbb{R}^{ly}$ tali che: $q = (q_1, ..., q_{l_x})$ e $r = \left(r_1, ..., r_{l_y}\right)$.

Definiamo ora due variabili aleatorie discrete $\hat{x}$ e $\hat{y}$ con massa di probabilità $P_{\hat{x}\hat{y}}(i, j)$ corrispondente alla probabilità che x e y appartengano ad un piccolo intervallo intorno a (q_i, r_j):

$$P_{\hat{x}\hat{y}}(i, j) = \int_{q_i}^{q_i+\Delta x} \int_{r_j}^{r_j+\Delta y} p_{xy}(x, y)\, dxdy \tag{2.78}$$

$$P_{\hat{x}}(i) = \int_{q_i}^{q_i+\Delta x} p_x(x)\, dx \qquad P_{\hat{y}}(j) = \int_{r_j}^{r_j+\Delta y} p_y(y)\, dy \tag{2.79}$$

$$\tag{2.80}$$

L'informazione mutua di $\hat{x}$ e $\hat{y}$ dalla definizione sarà:

$$I(\hat{x},\hat{y}) = \sum_{i=1}^{l_x}\sum_{j=1}^{l_y} P_{\hat{x}\hat{y}}(i,j)\log\left(\frac{P_{\hat{x}\hat{y}}(i,j)}{P_{\hat{x}}(i)\,P_{\hat{y}}(j)}\right) \tag{2.81}$$

Si dimostra facilmente che ([29]):

$$I(\hat{x},\hat{y}) \xrightarrow[\Delta x,\Delta y\to 0]{} I(x,y)$$

Vediamo ora sotto quali ipotesi sia possibile approssimare l'informazione mutua discreta delle due variabili aleatorie con l'informazione mutua di due vettori aleatori di dimensione opportuna.

Definiamo una mappa $\varphi : \mathbb{R} \to \mathbb{R}^p$. Avremo quindi $\varphi(\hat{x}) = \hat{\mathbf{x}}$ e $\varphi(\hat{y}) = \hat{\mathbf{y}}$ con $\hat{\mathbf{x}}$ e $\hat{\mathbf{y}}$ appartenenti a $\mathbb{R}^p$. Definiamo φ in modo che $\hat{x} = i$ sia equivalente a $\begin{cases} (\hat{\mathbf{x}})_i = 1 \\ (\hat{\mathbf{x}})_{i\neq j} = 0 \end{cases}$.

Definiamo due funzioni

$$k_i(x) = \begin{cases} 1 & x \in [q_i, q_i + \Delta x) \\ 0 & altrove \end{cases} \qquad k_j(y) = \begin{cases} 1 & y \in [r_j, r_j + \Delta y) \\ 0 & altrove \end{cases}$$

Ricaviamo ora delle relazioni utili.

$$E\{k_i(x)\} = E\{(\hat{\mathbf{x}})_i\} = \int k_i(x)\,p_x(x)\,dx = P_{\hat{x}}(i) \tag{2.82}$$

$$E\{k_i(x)\,k_j(y)\} = E\left\{(\hat{\mathbf{x}})_i\,(\hat{\mathbf{y}})_j\right\} = \\ = \int\int k_i(x)\,k_j(y)\,p_{xy}(x,y)\,dxdy = P_{\hat{x}\hat{y}}(i,j) \tag{2.83}$$

ponendo $x = y$ avremo $k_i(x) = k_i(y)$ e $p_{xy}(x,y) = p_x(x)\delta(x-y)$.

$$E\{k_i(x)\,k_j(x)\} = E\left\{\left(\hat{\mathbf{x}}\hat{\mathbf{x}}^t\right)_{ij}\right\} = \\ = \int\int k_i(x)\,k_j(x)\,p_x(x)\,\delta(x-y)\,dxdy = \begin{cases} P_{\hat{x}}(i) & i = j \\ 0 & altrove \end{cases} \tag{2.84}$$

Dalle relazioni appena trovate, possiamo scrivere:

$$E\left\{\hat{\mathbf{x}}\hat{\mathbf{y}}^t\right\} = P_{xy} \quad E\left\{\hat{\mathbf{x}}\right\} = p_x \quad E\left\{\hat{\mathbf{x}}\hat{\mathbf{x}}^t\right\} = D_x = diag\left(p_x\right) \tag{2.85}$$

dove con P_{xy} abbiamo indicato la matrice con coefficienti $P_{\hat{x}\hat{y}}\left(i,j\right)$ e con p_x il vettore con componenti $P_{\hat{x}}\left(i\right)$. Le ultime due relazioni valgono anche per il vettore $\hat{\mathbf{y}}$. In ultimo avremo:

$$C_{xy} = P_{xy} - p_x p_y^t \quad C_{xx} = D_x - p_x p_x^t \quad C_{yy} = D_y - p_y p_y^t \tag{2.86}$$

Definiamo ora due vettori aleatori Gaussiani, $\mathbf{x}_G$ e $\mathbf{y}_G$, con la stessa struttura di covarianza di $\hat{\mathbf{x}}$ e $\hat{\mathbf{y}}$. Da quanto detto prima avremo che:

$$\begin{aligned} I^G\left(x_G, y_G\right) &= -\frac{1}{2}\log\left(\frac{\det C}{\det C_{xx}\det C_{yy}}\right) = \\ &= -\frac{1}{2}\log\left(\prod_i\left(1-\rho_i\right)\right) \end{aligned} \tag{2.87}$$

dove i ρ_i sono gli autovalori del problema

$$\begin{pmatrix} 0 & P_{xy} - p_x p_y^t \\ \left(P_{xy} - p_x p_y^t\right)^t & 0 \end{pmatrix}\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \rho\begin{pmatrix} D_x - p_x p_x^t & 0 \\ 0 & D_y - p_y p_y^t \end{pmatrix}\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \tag{2.88}$$

(per i passaggi matematici consultare l'appendice B.2)

La relazione tra $I^G\left(x_G, y_G\right)$ e $I\left(\hat{x},\hat{y}\right)$ è data dal seguente lemma.

Lemma 2.1. *Se $P_{\hat{x}\hat{y}}\left(i,j\right)$ può essere approssimata come $P_{\hat{x}}\left(i\right)P_{\hat{y}}\left(j\right)\left(1+\varepsilon_{ij}\right)$ con ε_{ij} piccolo (espansione intorno all'indipendenza), allora lo sviluppo di Taylor arrestato al secondo ordine di $I\left(\hat{x},\hat{y}\right)$ è uguale allo sviluppo di Taylor arrestato al secondo ordine di $I^G\left(x_G, y_G\right)$.*

Una dimostrazione di questo lemma si trova nell'appendice (B.3)

Vediamo ora come cambiano le cose se invece di usare una mappa $\varphi : \mathbb{R} \to \mathbb{R}^p$ prendessimo una mappa $\phi : \mathbb{R} \to \mathcal{F}$ dove $\mathcal{F}$ è un RKHS. Supponiamo di utilizzare un kernel gaussiano $k_\sigma\left(x,y\right) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(x-y)^2}{2\sigma^2}\right)$. Ricordiamo anche che il kernel gaussiano soddisfa la seguente proprietà: $\delta\left(x-y\right) = \lim_{\sigma\to 0} k_\sigma\left(x,y\right)$. Definiamo una griglia $\left(q_i, r_j\right)$ come in precedenza e indichiamo con α_{l_x}

e β_{l_y} due vettori di funzioni definiti nel seguente modo:

$$\alpha_{l_x} = \begin{pmatrix} k(x,q_1) & \cdots & k(x,q_{l_x}) \end{pmatrix} \quad \beta_{l_y} = \begin{pmatrix} k(y,r_1) & \cdots & k\left(y,r_{l_y}\right) \end{pmatrix} \tag{2.89}$$

Notiamo che, se la griglia diventa infinitamente fine, i due vettori di funzioni tendono alla continuità nella seconda variabile riottenendo il kernel $k(\cdot,\cdot)$. In altre parole stiamo usando le funzioni dei vettori α_{l_x} e β_{l_y} come generatori di due spazi a dimensione finita

$$\mathcal{F}\{q_i\} = Span\left(\{k(x,q_i)\}_{i=1}^{l_x}\right) \quad \mathcal{F}\{r_j\} = Span\left(\{k(y,r_j)\}_{j=1}^{l_y}\right) \tag{2.90}$$

tali che

$$\mathcal{F}\{q_i\}, \mathcal{F}\{r_j\} \xrightarrow[\Delta x, \Delta y \to 0]{} \mathcal{F} \tag{2.91}$$

cioè, quando la griglia diventa infinitamente fine, i due spazi tendono a $\mathcal{F}$.

Procediamo ora esattamente come abbiamo già fatto nel paragrafo precedente, definiamo cioè due vettori aleatori gaussiani $\mathbf{x}_G$ e $\mathbf{y}_G$ con la stessa struttura di covarianza delle variabili $\phi_q(x)$ e $\phi_r(y)$ dove ϕ_q e ϕ_r sono mappe tali che:

$$\begin{aligned} \phi_q : \mathbb{R} &\to \mathcal{F}\{q_i\} & \qquad \phi_r : \mathbb{R} &\to \mathcal{F}\{r_j\} \\ x &\mapsto f_1 & \qquad y &\mapsto f_2 \end{aligned}$$

Calcoleremo poi l'informazione mutua $I^G(\mathbf{x}_G, \mathbf{y}_G)$ e vedremo in che modo questa approssimi l'informazione mutua $I(x,y)$ tra le variabili di partenza. Per calcolare la $I^G(\mathbf{x}_G, \mathbf{y}_G)$ abbiamo bisogno dello spettro del problema agli autovalori del CCA, devo cioè ricondurmi al problema (2.88). Cerchiamo la struttura di covarianza di f_1 e f_2. In particolare ho bisogno di conoscere tre quantità: $\operatorname{cov}\{f_1, f_2\}$, $\operatorname{var}\{f_1\}$ e $\operatorname{var}\{f_2\}$. Conoscendo queste grandezze posso interpretarle come se fossero le statistiche del secondo ordine di due variabili gaussiane nel RKHS considerato. Detto questo andiamo a calcolarci il rapporto $\frac{\operatorname{cov}\{f_1,f_2\}}{\operatorname{var}\{f_1\}^{1/2}\operatorname{var}\{f_2\}^{1/2}}$ e da qui passare allo spettro del CCA.

$$\begin{aligned}
\operatorname{cov}\{f_1, f_2\} &= cov\left\{\langle f_1, \phi_q(x)\rangle, \langle f_2, \phi_r(y)\rangle\right\} = \\
&= E\{f_1(x), f_2(y)\} - E\{f_1(x)\} E\{f_2(y)\} = \\
&= \sum_{i=1}^{l_x}\sum_{j=1}^{l_y} \alpha_1^i \alpha_2^j E\{k(x, q_i), k(y, r_j)\} \Delta x \Delta y - \\
&- \sum_{i=1}^{l_x}\sum_{j=1}^{l_y} \alpha_1^i \alpha_2^j E\{k(x, q_i)\} E\{k(y, r_j)\} \Delta x \Delta y = \\
&= \alpha_1^t E\left\{\alpha_{l_x}\beta_{l_y}^t\right\} \alpha_2 \Delta x \Delta y - \alpha_1^t E\{\alpha_{l_x}\} E\left\{\beta_{l_y}^t\right\} \alpha_2 \Delta x \Delta y = \\
&= \alpha_1^t \left(E\left\{\alpha_{l_x}\beta_{l_y}^t\right\} - E\{\alpha_{l_x}\} E\left\{\beta_{l_y}^t\right\}\right) \alpha_2 \Delta x \Delta y \quad (2.92)
\end{aligned}$$

$$\begin{aligned}
&\operatorname{var}\{f_1\} = E\left\{f_1^2(x)\right\} - E^2\{f_1(x)\} = \\
&= \sum_{i=1}^{l_x}\sum_{j=1}^{l_x} \alpha_1^i \alpha_1^j E\{k(x, q_i) k(x, q_j)\} - \sum_{i=1}^{l_x}\sum_{j=1}^{l_x} \alpha_1^i \alpha_1^j E\{k(x, q_i)\} E\{k(x, q_j)\} = \\
&= \alpha_1^T E\left\{\alpha_{l_x}\alpha_{l_x}^T\right\} \alpha_1 \Delta_x^2 - \alpha_1^T E\{\alpha_{l_x}\} E\left\{\alpha_{l_x}^T\right\} \alpha_1 \Delta_x^2 = \\
&= \alpha_1^T \left(E\left\{\alpha_{l_x}\alpha_{l_x}^T\right\} - E\{\alpha_{l_x}\} E\left\{\alpha_{l_x}^T\right\}\right) \alpha_1 \Delta_x^2 \quad (2.93)
\end{aligned}$$

quest'ultima relazione vale anche per $\operatorname{var}\{f_2\}$.

Avremo quindi che:

$$\frac{\alpha_1^t \left(E\left\{\alpha_{l_x}\beta_{l_y}^t\right\} - E\{\alpha_{l_x}\} E\left\{\beta_{l_y}^t\right\}\right) \alpha_2}{\left(\alpha_1^t \left(E\left\{\alpha_{l_x}\alpha_{l_x}^T\right\} - E\{\alpha_{l_x}\} E\left\{\alpha_{l_x}^T\right\}\right) \alpha_1\right)^{1/2} \left(\alpha_2^t \left(E\left\{\alpha_{l_y}\alpha_{l_y}^T\right\} - E\{\alpha_{l_y}\} E\left\{\alpha_{l_y}^T\right\}\right) \alpha_2\right)^{1/2}} \quad (2.94)$$

Se noi ora riuscissimo a dimostrare le relazioni che legano le seguenti grandezze

$$P_{xy} \leftrightarrow \Delta x \Delta y E\left\{\alpha_{l_x}\beta_{l_y}^t\right\} \quad D_x \leftrightarrow \Delta x^2 E\left\{\alpha_{l_x}\alpha_{l_x}^t\right\} \quad p_x \leftrightarrow \Delta x E\{\alpha_{l_x}\} \quad (2.95)$$

(e le corrispondenti relazioni per le grandezze in y) ci saremmo ricondotti al problema (2.88). Questo concluderebbe la dimostrazione. Basterebbe infatti ripercorrere gli stessi passi fatti

prima e in particolare, applicare il Lemma 2.1 per ottenere il risultato.

Cominciamo con

$$\left(E\left\{\alpha_{l_x}\beta_{l_y}^t\right\}\right)_{ij} =$$
$$= E\left\{k\left(x,q_i\right)k\left(y,r_j\right)\right\} = \int\int k\left(x-q_i\right)k\left(y-r_j\right)p_{xy}\left(x,y\right)dxdy =$$
$$\left(k\left(x\right)k\left(y\right) * p_{xy}\left(x,y\right)\right)\left(q_i,r_j\right) \quad (2.96)$$

Questa convoluzione, valutata in (q_i, r_j), rappresenta una massa di probabilità in quanto il kernel gaussiano soddisfa la proprietà di normalizzazione ($\int k\left(x\right)dx = 1$).

In maniera analoga ricaviamo

$$\left(E\left\{\alpha_{l_x}\alpha_{lx}^t\right\}\right)_{ij} =$$
$$= E\left\{k\left(x,q_i\right)k\left(x,q_j\right)\right\} = \int\int k\left(x-q_i\right)k\left(x-q_j\right)p_x\left(x\right)dx$$
$$\simeq \begin{cases} \left(k^2\left(x\right) * p_x\left(x\right)\right)\left(q_i\right) & i=j \\ 0 & altrove \end{cases} \quad (2.97)$$

In questa approssimazione abbiamo supposto σ sufficientemente piccola. Notiamo che questa non rappresenta una massa di probabilità in quanto $\int k^2\left(x\right)dx = \frac{1}{2\sigma\sqrt{\pi}}$. Osserviamo però che:

$$k^2\left(x\right) = \frac{1}{2\pi\sigma^2}e^{-\frac{x^2}{\sigma^2}} = \frac{1}{2\sigma\sqrt{\pi}} \cdot \frac{1}{\sqrt{\pi}\sigma}e^{-\frac{x^2}{\sigma^2}} = \frac{1}{2\sigma\sqrt{\pi}}k'\left(x\right) \quad (2.98)$$

dove $k'\left(x\right)$ tende a una delta di Dirac quando σ tende a 0. Possiamo quindi scrivere:

$$\left(E\left\{\alpha_{l_x}\alpha_{lx}^t\right\}\right)_{ij} \simeq \begin{cases} 2\sigma\sqrt{\pi}\left(k'\left(x\right) * p_x\left(x\right)\right)\left(q_i\right) & i=j \\ 0 & altrove \end{cases} \quad (2.99)$$

che rappresenta una densità di probabilità scalata della costante $2\sigma\sqrt{\pi}$. In ultimo ci resta

$$\left(E\left\{\alpha_{l_x}\right\}\right)_i = \int k\left(x-q_i\right)p_x\left(x\right)dx = \left(k\left(x\right) * p_x\left(x\right)\right)\left(q_i\right) \quad (2.100)$$

Come abbiamo già detto, se $\sigma \to 0$ allora $k(x) \to \delta(x)$. Dalle relazioni precedenti avremo quindi che:

$$\left(E\left\{\alpha_{l_x}\beta_{l_y}^t\right\}\right)_{ij} \xrightarrow[\sigma\to 0]{} p_{xy}(q_i, r_j) \quad (E\{\alpha_{l_x}\})_i \xrightarrow[\sigma\to 0]{} p_x(q_i) \tag{2.101}$$

Passando ora al limite Δx e Δy otterremo le ultime relazioni che ci permetteranno di concludere la dimostrazione. Infatti con $\Delta x, \Delta y \to 0$ avremo:

$$P_{\hat{x}\hat{y}}(i,j) = \int_{q_i}^{q_i+\Delta x} \int_{r_j}^{r_j+\Delta x} p_{xy}(x,y)\,dxdy \simeq \Delta x \Delta y p_{xy}(q_i, r_j) \tag{2.102}$$

$$P_{\hat{x}}(i) = \int_{q_i}^{q_i+\Delta x} p_x(x)\,dx \simeq \Delta x p_x(q_i) \tag{2.103}$$

In conclusione possiamo affermare che valgono le seguenti relazioni:

$$\begin{aligned} P_{xy} &\simeq \Delta x \Delta y E\left\{\alpha_{l_x}\beta_{l_y}^t\right\} \\ \frac{\Delta x}{2\sigma\sqrt{\pi}} D_x &\simeq \Delta x^2 E\left\{\alpha_{l_x}\alpha_{l_x}^t\right\} \\ p_x &\simeq \Delta x E\{\alpha_{l_x}\} \end{aligned} \tag{2.104}$$

Questo conclude la dimostrazione.

L'ultima cosa che rimane da fare è trovare una stima dai dati di $E\left\{\alpha_{l_x}\beta_{l_y}^t\right\}$, $E\left\{\alpha_{l_x}\alpha_{l_x}^t\right\}$ e $E\{\alpha_{l_x}\}$, riagganciandoci quindi al problema KGV già risolto. Definiamo:

$$K_1 = \begin{pmatrix} k(x_1,q_1) & k(x_1,q_2) & \cdots & k(x_1,q_{lx}) \\ k(x_2,q_1) & & & \vdots \\ \vdots & & & \vdots \\ k(x_N,q_1) & \cdots & \cdots & k(x_N,q_{l_x}) \end{pmatrix} \tag{2.105}$$

$$K_2 = \begin{pmatrix} k(y_1,r_1) & k(y_1,r_2) & \cdots & k(y_1,r_{ly}) \\ k(y_2,r_1) & & & \vdots \\ \vdots & & & \vdots \\ k(y_N,r_1) & \cdots & \cdots & k\left(y_N,r_{l_y}\right) \end{pmatrix} \tag{2.106}$$

dove $\{x_1,\ldots,x_N\}$ e $\{y_1,\ldots,y_N\}$ sono i dati. Dobbiamo sostituire i valori medi presenti in $\left(E\left\{\alpha_{l_x}\beta_{l_y}^t\right\}\right)_{ij} = E\{k(x,q_i)\,k(y,r_j)\}$,

$\left(E\left\{\alpha_{l_x}\alpha_{lx}^t\right\}\right)_{ij} = E\left\{k\left(x,q_i\right)k\left(x,q_j\right)\right\}$ e $\left(E\left\{\alpha_{l_x}\right\}\right)_i$ con le relative medie campionarie. Partiamo con

$$\begin{aligned}\left(E\left\{\alpha_{l_x}\beta_{l_y}^t\right\}\right)_{ij} &= E\left\{k\left(x,q_i\right)k\left(y,r_j\right)\right\} \simeq \\ &\simeq \frac{1}{N}\sum_{n=1}^{N} k\left(x_n,q_i\right)k\left(y_n,r_j\right)\end{aligned} \tag{2.107}$$

Le altre possono essere ricavate in maniera analoga. Usando la notazione matriciale in cui le medie sono date dai prodotti riga-colonna, e fissando $\Delta x = \Delta y = \Delta$ riscriviamo direttamente che:

$$P_{xy} - p_x p_y^t = \frac{\Delta\Delta}{N}\left(K_1K_2 - \frac{1}{N}K_1\mathbf{1}K_2\right) = \frac{\Delta\Delta}{N}K_1ZK_2 \tag{2.108}$$

$$\frac{\Delta}{2\sqrt{\pi}}D_x - p_x p_x^t = \frac{\Delta^2}{N}\left(K_1K_1 - \frac{1}{N}K_1\mathbf{1}K_1\right) = \frac{\Delta\Delta}{N}K_1ZK_1 \tag{2.109}$$

dove Z è la matrice già definita nell'appendice (B.1) come $Z = \left(I - \frac{1}{N}\mathbf{1}\right)$. Usando le relazioni appena trovate possiamo riscrivere il problema (2.88) come:

$$\begin{pmatrix} 0 & K_1ZK_2 \\ K_2ZK_1 & 0 \end{pmatrix}\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \rho\begin{pmatrix} K_1ZK_1 & 0 \\ 0 & K_2ZK_2 \end{pmatrix}\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \tag{2.110}$$

Mostriamo ora come questo problema sia lo stesso di (2.63) quello cioè ricavato per la stima dello spettro del KCCA .

Pre-moltiplichiamo e post-moltiplichiamo entrambi i membri della (2.110) per una matrice a blocchi $\begin{pmatrix} Z & 0 \\ 0 & Z \end{pmatrix}$, avremo quindi

$$\begin{pmatrix} 0 & ZK_1ZK_2Z \\ ZK_2ZK_1Z & 0 \end{pmatrix}\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \rho\begin{pmatrix} ZK_1ZK_1Z & 0 \\ 0 & ZK_2ZK_2Z \end{pmatrix}\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \tag{2.111}$$

Osservando che $Z^2 = Z$ e che $\tilde{K} = ZKZ$ come mostrato nell'appendice (B.1), avremo:

$$\begin{pmatrix} 0 & \tilde{K}_1\tilde{K}_2 \\ \tilde{K}_2\tilde{K}_1 & 0 \end{pmatrix}\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \rho\begin{pmatrix} \tilde{K}_1\tilde{K}_1 & 0 \\ 0 & \tilde{K}_2\tilde{K}_2 \end{pmatrix} \tag{2.112}$$

Quest'ultima relazione è esattamente identica a (2.63). Per le valutazioni sulla correttezza di questa stima e su come modificarla, si rimanda alla sezione precendente in cui se ne è già discusso.

Conclusioni sulla contrast function KGV

Come abbiamo potuto vedere, la tecnica KGV ha ancora molti punti oscuri e ancora molto lavoro deve essere fatto per migliorarla. L'applicazione di questa tecnica al problema TCA rappresenta un esperimento. Ci sono infatti alcuni problemi che la contrast function KGV non riesce a risolvere. Il più importante di tutti è il fatto che la tecnica KGV non preserva l'indipendenza condizionata ma solo quella marginale. In altre parole: se il vettore aleatorio di partenza x fattorizza in una foresta T nel senso visto nella sezione (2.2.2), il vettore $\phi(x)$ (cioè la proiezione di x in un RKHS) può non fattorizzare più in una foresta, in quanto la struttura di dipendenza condizionata potrebbe venire distrutta dal mappaggio. Recenti lavori ([20]) stanno mettendo a punto delle tecniche per risolvere questo problema.

Capitolo 3

Separazione delle sorgenti in astrofisica

La cosmologia è la scienza che studia le origini e l'evoluzione dell'universo. Per molti secoli si è pensato che l'Universo fosse statico, si credeva cioè che su larga scala nulla si muovesse. Questa concezione è stata messa in crisi dalle osservazioni di Edwin Hubble nel 1929. Lo scienziato, misurando lo spostamento Doppler della luce proveniente dalle galassie, scoprì infattì che quest'ultime si stavano allontanando reciprocamente una dall'altra. Hubble ipotizzò quindi che l'Universo si stesse espandendo a velocità costante. Anche se molto imprecise, queste scoperte misero in crisi il modello stazionario. Negli anni successivi gli scienziati svilupparono le conclusioni a cui era giunto Hubble e misero a punto quella che oggi è conosciuta come *teoria del Big Bang*. Lo sforzo degli scienziati negli ultimi anni è stato quello di approntare delle missioni spaziali con lo scopo di cercare verifiche sperimentali ai modelli teorici. Il satellite Planck dell'Agenzia Spaziale Europea è la più ambiziosa di queste missioni.

3.1 Una prova del Big Bang: *Cosmic Microwave Background*

Il problema degli astrofisici è stato in primo luogo quello di trovare una prova sperimentale del Big Bang, una prova che fosse possibile osservare ora dopo milioni di anni. Questa prova è stata

trovata nel 1964 quasi per caso. In quell'anno infatti due ricercatori rivelarono una radiazione che proveniva da ogni punto del cielo (non era quindi generata da una sorgente compatta), una radiazione che sembrava occupare con la stessa intensità l'intero Universo. Questa radiazione è detta *Cosmic Microwave Background* (CMB). Cerchiamo di capire di cosa si tratta e perchè può essere considerata una prova del Big Bang, descrivendo a grandi linee la nascita dell'Universo secondo appunto questa teoria.

- 10^{-43} sec. Questa quantità temporale, detta *tempo di Planck*, è la più piccola che ha senso considerare per le attuali teorie scientifiche. Quello che è avvenuto prima di tale istante non può in alcun modo essere predetto nè con la teoria della relatività, nè con la meccanica quantistica. La temperatura era dell'ordine di 10^{32}K.
- 10^{-35} sec. Inizia la fase inflazionaria. La teoria inflazionaria, sviluppata da Linde, prevede un'espansione esponenziale dell'Universo sotto la spinta dell'energia proveniente dalla formazione e dal conseguente annichilimento di particelle virtuali. La temperatura si abbassa fino a valori di 10^{27}, 10^{28}K ad un tempo di 10^{-35} sec dal Big Bang. Questo abbassamento di temperatura porta ad una rottura spontanea della simmetria delle forze fondamentali.
- 10^{-33} sec, fine della fase inflazionaria. La temperatura rimane intorno ai 10^{27}K. Avviene quel fenomeno chiamato bariogenesi: coppie di particella-antiparticella si creano e si annichiliscono istantaneamente dando origine a un gran numero di fotoni. Durante questo processo però vengono create più particelle che antiparticelle, quindi non tutta la materia si annichilisce.
- 0.0001 sec. L'Universo si sta espandendo e raffreddando, cresce la disomogeneità tra materia e antimateria che va lentamente scomparendo. Si formano protoni e neutroni. La temperatura è di circa 10^{13}K.
- 100 sec. La temperatura è scesa a circa 10^{9}K. Compaiono i primi elettroni, mentre protoni e neutroni si combinano insieme dando origine ai primi nuclei di deuterio e di elio.
- Un mese dopo il Big Bang si fissa lo spettro di quello che verrà poi chiamato CMB *Cosmic Microwave Background*. Lo spettro è esattamente quello del corpo nero. Il CMB è

una radiazione formata da fotoni che a causa dell'alta densità e delle iterazioni tra particelle, non riesce ancora a staccarsi dalla materia che si sta formando.

- 380000 anni dopo il Big Bang. La temperatura scende a 3000K, gli elettroni si legano ai protoni per formare atomi di idrogeno. L'Universo diventa "trasparente", nel senso che i fotoni che costituiscono il CMB riescono a liberarsi dalle iterazioni con le altre particelle e a viaggiare libero nello spazio. I fotoni però non vengono scatterati in maniera omogenea verso tutte le direzioni, ma presentano delle anisotropie. L'importanza cruciale del CMB sta proprio in queste anisotropie che caratterizzano l'Universo primordiale.
- 100-200 milioni di anni dopo il Big Bang cominciano a formarsi le prime stelle.
- 13.7 bilioni di anni dopo il Big Bang: stato attuale dell'Universo.

La temperatura media del CMB è di 2.73K. Le anisotropie sono variazioni di una parte su milione da questa temperatura media. Su vasta scala la struttura dell'Universo è omogenea, mentre su piccola scala presenta evidenti disomogeneità (ad esempio galassie, ammassi di galassie ecc). Le anisotropie sono generate proprio da quei piccoli grumi di materia che avrebbero poi dato origine alle strutture su piccola scala.

Da quanto detto, risulta evidente che una misura quanto più precisa possibile del CMB e soprattutto delle sue anisotropie sia necessaria per confermare le attuali teorie cosmologiche.

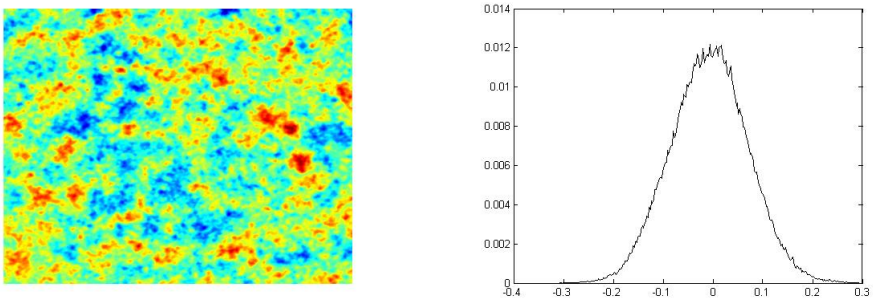


Figura 3.1: Una patch di CMB simulato e il suo istogramma

3.2 La missione Planck

Nel 1992 la missione COBE misurò le fluttuazioni del CMB con una risoluzione di circa 10° e una sensitività $\Delta T/T$ di circa 10^{-5}. Il satellite COBE era dotato di tre canali frequenziali con frequenze di centro banda pari a 31.5, 53, 90 GHz. Nel 1998 venne avviata, grazie a una collaborazione tra Italia e Stati Uniti, la missione Boomerang. Tramite un pallone aereostatico vennero effettuate misure più precise delle anisotropie con risoluzioni angolari di circa 1°. Più recentemente, nel 2003, la missione WMAP rifece le stesse misure con una risoluzione di 15 arcmin e con una precisione simile a quella del COBE. La differenza però sta nel maggior numero di canali frequenziali: il satellite WMAP ne aveva cinque (K, Ka, Q, V, W) con frequenze di centro banda di 22, 30, 40, 60, 94 GHz (figura 3.2). La missione Planck ha l'obiettivo di migliorare queste misurazioni. La risoluzione sarà infatti minore di 5 arcmin e la sensitività sarà dell'ordine di 10^{-6}. Il satellite Planck coprirà poi una gamma di frequenze molto maggiori (da 30 GHz a 857 GHz) (figura 3.3) rispetto ai due satelliti precedenti. Se infatti COBE e WMAP avevano rispettivamente tre e cinque canali, il satellite Planck ne monterà nove. Questo permetterà di separare il CMB dalle altre sorgenti astrofisiche in maniera più precisa, ma anche di raccogliere una grande quantità di informazioni su quest'ultime. I rivelatori ad alta frequenza (da 100 a 857 GHz) aprono una finestra nuova sull'Universo. Nessuna missione precedente infatti aveva mai acquisito dati in questa gamma di frequenze. Daremo più avanti una breve descrizione delle possibili sorgenti astrofisiche.

3.2.1 Caratteristiche tecniche

Il satellite Planck usa un telescopio di 1,5 metri di diametro con struttura a offset. Sul piano focale sono montati due gruppi di rivelatori che coprono nove bande frequenziali. Il gruppo di rivelatori a bassa frequenza (30, 44 e 70 GHz) sono del tipo HEMT e sono raffreddati a una temperatura di circa 20K. Il secondo gruppo copre le frequenze più alte (100, 143, 217, 353, 545, 857 GHz), sono di tipo bolometrico e sono raffreddati fino a 0.1K. Il satellite Planck sarà posizionato in un'orbita nella quale il telescopio riuscirà sempre a puntare nella direzione opposta a quella del Sole

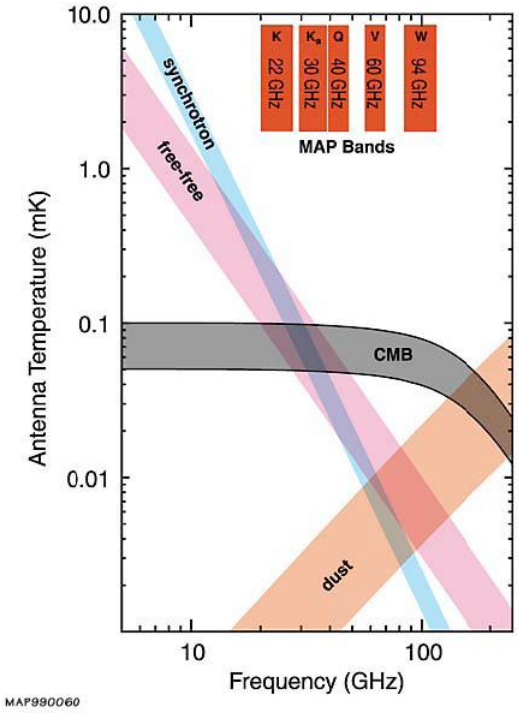


Figura 3.2: Ampiezza delle sorgenti in confronto alla frequenza nella banda del WMAP

per diminuire il più possibile l'interferenza causata dalla stella. Il satellite eseguirà due scansioni complete della volta celeste e impiegherà circa 14 mesi.

3.3 Le altre sorgenti

Come abbiamo già detto l'obiettivo della missione Planck è quello di analizzare con precisione il CMB. Nello spazio però sono presenti molte altre sorgenti che emmettono una radiazione elettromagnetica la cui frequenza dipende dalla temperatura della sorgente stessa. Il telescopio quindi non riceverà solo il segnale proveniente dal CMB ma una mistura lineare di segnali. Bisognerà quindi, utilizzando tutta l'informazione proveniente dai canali

Telescopio	1.5 m di diametro, direzione di vista dell'offset: 85° dall'asse di rotazione								
Strumenti	LFI			HFI					
Tipo di rivelatori	HEMT LNA arrays			bolometers arrays					
Temperatura dei rivelatori	$\sim$ 20K			0.1K					
Frequenza di centro banda (GHz)	30	44	70	100	143	217	353	545	857
Numero di rivelatori	4	6	12	8	12	12	12	4	4
Bandwidth (GHz)	6	8.8	14	33	47	72	116	180	283
Risoluzione angolare (FWHM, arcmin)	33	24	14	9.5	7.1	5	5	5	5
ΔT/T media per pixel (unità 10^{-6})	2.0	2.7	4.7	2.5	2.2	4.8	14.7	147	6700

Tabella 3.1: Caratteristiche tecniche del satellite Planck ([30])

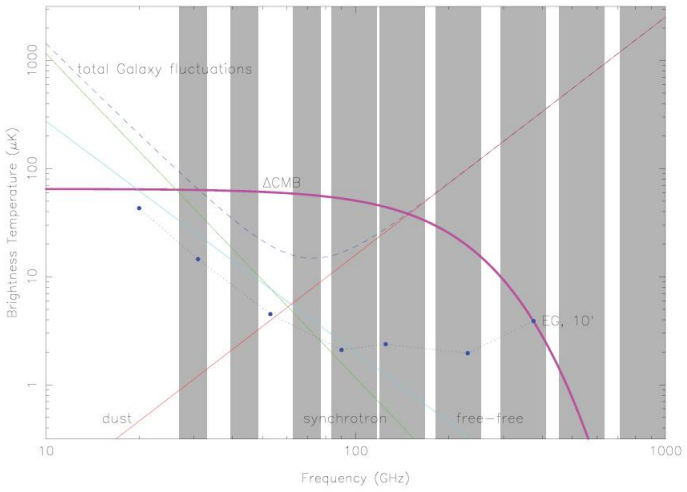


Figura 3.3: Ampiezza delle sorgenti in confronto alla frequenza nella banda del satellite Planck

frequenziali, separare i contributi delle varie sorgenti dal CMB. Diamo ora una breve descrizione delle possibili sorgenti.

3.3.1 Galactic dust

In una galassia, oltre agli oggetti "compatti" come stelle e pianeti, c'è una grande quantità di materia simile a un gas. La temperatura di questo gas è di circa 20K. Il contributo di questa sorgente si fa sentire soprattutto alle alte frequenze analizzate dal satellite Planck, mentre è molto basso alle frequenze di emissione del CMB (intorno ai 100 GHz). Recuperare le informazioni su questa sorgente può essere molto importante oltre che per ripulire le mappe del CMB, anche per conoscere meglio la struttura della nostra Galassia.

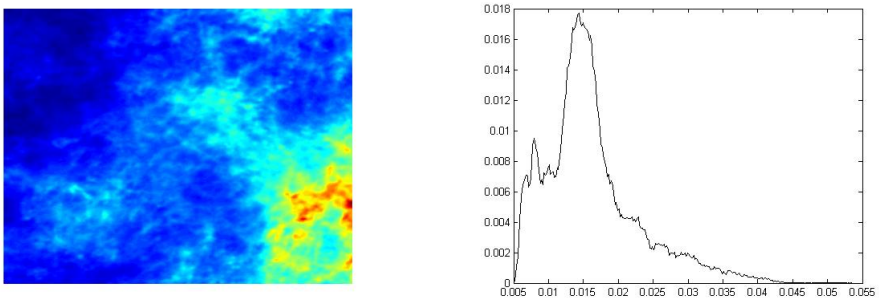


Figura 3.4: Una patch di dust simulato e il suo istogramma

3.3.2 Synchrotron

Il synchrotron è un altro tipo di radiazione proveniente dalla nostra Galassia. Questa componente porta informazioni sulla struttura del campo magnetico galattico, sulla distribuzione spaziale e energetica degli elettroni relativistici e sulla loro densità. Può essere utile per misurare le variazioni della densità degli elettroni e del campo magnetico prodotte dalle esplosioni delle supernova. Il contributo del synchrotron si fa sentire soprattutto a basse frequenze.

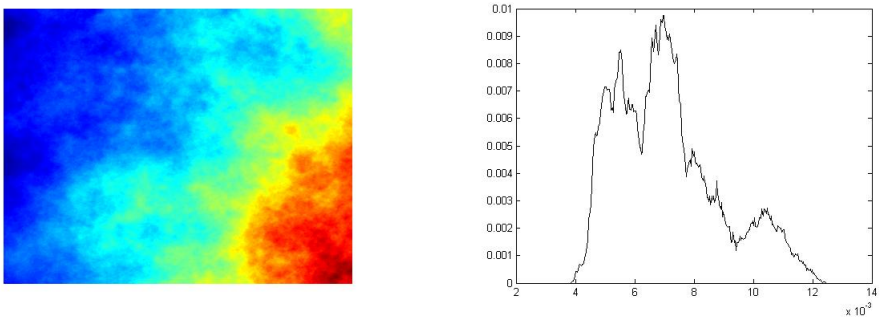


Figura 3.5: Una patch di synchrotron simulato e il suo istogramma

3.3.3 Free-Free Emission

Con il nome di free-free emission viene indicato il processo di emissione di un fotone da parte di un elettrone dovuta a un'interazione elettrostatica con un'altra particella carica. In altre parole quando un elettrone viene deviato da un nucleo atomico può emettere o assorbire un fotone. La free-free emission può essere osservato sia nelle nebulose galattiche in cui è presente gas ionizzato sia nei cluster di galassie in cui è presente del gas "intracluster" ad elevata temperatura ($8.8 \cdot 10^7$K). Lo studio della free-free emission può dare informazioni utili su come la distribuzione della massa e della temperatura delle galassie varia in funzione della distanza dal centro galattico.

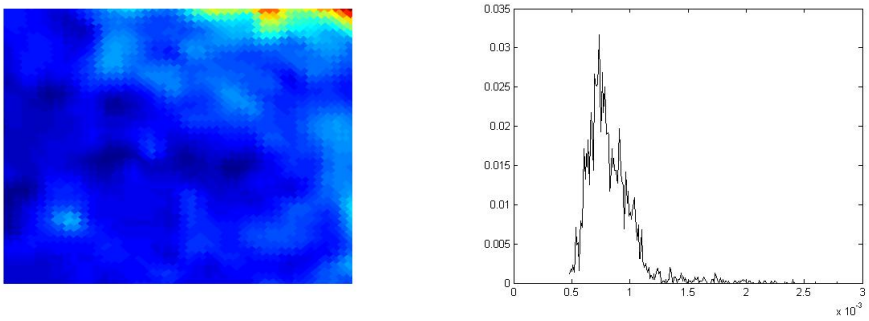


Figura 3.6: Una patch di free-free emission simulato e il suo istogramma

3.3.4 Rumore degli strumenti

Ogni strumento di acquisizione di dati porta con se una inevitabile quantità di rumore. Un rumore di questo tipo risulta essere perfettamente noto in quanto può essere studiato direttamente sugli strumenti. Anche nel caso del satellite Planck disponiamo di una mappa dettagliata del rumore. In particolare questo rumore risulta gaussiano ma spazio-variante, cioè ogni pixel presenta una varianza del rumore diversa.

Capitolo 4

Simulazioni

In questa sezione presenteremo i risultati ottenuti con dati sintetici. Useremo due tipi di dati: i primi saranno generati col Matlab e rappresentano delle distribuzioni che fattorizzano in un albero ben definito. Il secondo tipo di dati sono invece dati astrofisici ottenuti partendo dalle conoscenze teoriche che si hanno sulle varie sorgenti. L'uso dei dati sintetici è utile per due motivi:

- confrontare i due metodi per la stima della contrast function (KDE e KGV).
- confrontare le prestazioni dell'algoritmo TCA con quelle ottenute sfruttando l'algoritmo FastICA, Multidimensional ICA, Topographic ICA e CorCA.

4.1 Divergenza di Amari

Per fare il confronto delle prestazioni abbiamo bisogno di una misura della correttezza della stima. Sia nell'algoritmo FastICA sia nel TCA la grandezza che stimiamo è la demixing matrix $\hat{W}$. Se la stima fosse perfetta, indicando con A la mixing matrix, avremo che $\hat{W} = A^{-1}$ o equivalentemente $\hat{W}A^{-1} = I$ dove I è la matrice identica. Questo però nel caso ideale. Definendo quindi una matrice $B = \hat{W}A^{-1}$, il nostro scopo sarà quello di "misurare" di quanto B si discosta dalla matrice identica. Come abbiamo visto però la stima $\hat{W}$ è affetta dall'invarianza alla permutazione e a fattori di scala. Vorremmo quindi che la nostra metrica non

tenesse in conto queste invarianze. Una metrica comunemente usata è quella di Amari così definita:

$$\mathcal{D}(B) = \sum_{i=1}^{n} \left(\frac{\sum_{j=1}^{n} |B_{ij}|}{\max_j |B_{ij}|} - 1 \right) + \sum_{j=1}^{n} \left(\frac{\sum_{i=1}^{n} |B_{ij}|}{\max_i |B_{ij}|} - 1 \right) \quad (4.1)$$

In particolare:

- $D(B) \geqslant 0$. L'uguaglianza vale se e solo se $B = I$
- Sia P una matrice di permutazione e Λ una matrice diagonale non singolare allora: $\mathcal{D}(B) = 0 \Rightarrow \mathcal{D}(B\Lambda P) = 0$, $\mathcal{D}(\Lambda P B) = 0$. Questo significa che la divergenza di Amari è invariante alla permutazione e a fattori di scala

4.2 Dati sintetici

In questa sezione mostreremo i risultati ottenuti con i dati sintetici generati in Matlab. Abbiamo realizzato due tipi di dati che fattorizzano in due diversi alberi.

Primo caso

Abbiamo generato un vettore aleatorio che fattorizza nell'albero in figura 4.1

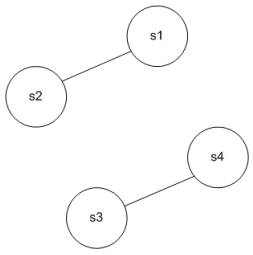


Figura 4.1: Modello dell'albero su cui fattorizzano i primi dati sintetici

Vediamo come sono state definite le componenti s_1, s_2, s_3, s_4. Indichiamo con:

- $u \sim \mathcal{U}$ una variabile aleatoria uniformemente distibuita

- $g \sim \Gamma$ una variabile aleatoria con distribuzione gamma
- $e \sim E$ una variabile aleatoria esponenziale
- $n \sim \mathcal{N}$ una variabile aleatoria gaussiana

Usando queste variabili aleatorie abbiamo poi costruito le componenti del vettore s nel seguente modo:

- $s_1 = u$
- $s_2 = u^2$
- $s_3 = g + e$
- $s_4 = g \cdot n$

Abbiamo poi premoltiplicato il vettore s per una matrice non singolare A ottenendo il vettore dei dati x.

Nella figura 4.2 mostriamo il confronto tra i due metodi di stima della contrast function KDE e KGV.

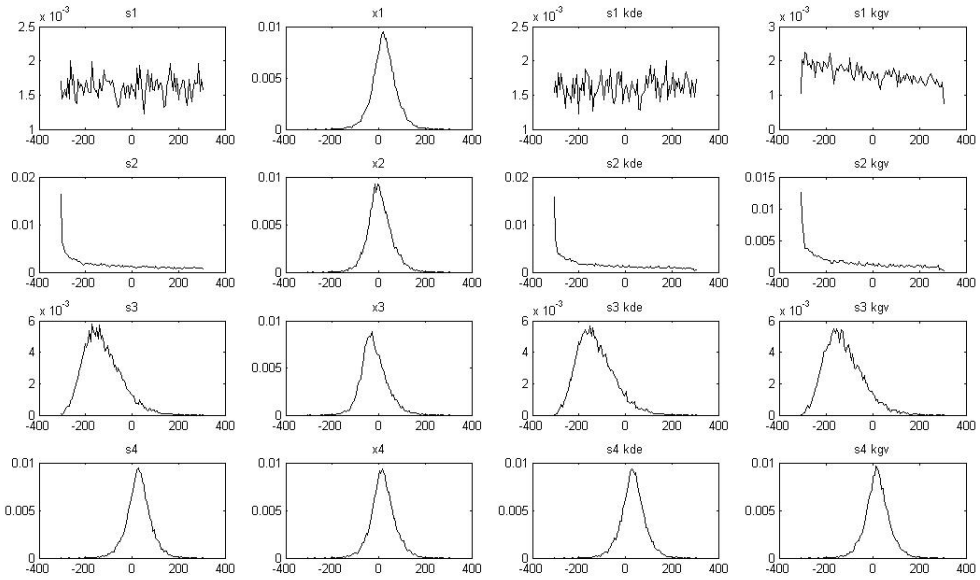


Figura 4.2: Nella prima colonna sono rappresentate le componenti del vettore s, la seconda colonna rappresenta le componenti del vettore dei dati x. Nelle ultime due colonne sono rappresentate le componenti stimate di s con i due metodi (KDE nella terza colonna e KGV nella quarta).

La divergenza di Amari nei due casi risulta:

- $\mathcal{D} = 2.3712$ per il metodo KDE
- $\mathcal{D} = 3.1082$ per il metodo KGV

Si può subito notare che la contrast function stimata con il metodo KDE risulta migliore. Per quanto riguarda la struttura di dipendenza, cioè il riconoscimento dell'albero, entrambi i metodi funzionano bene.

Nella figura 4.3 faremo il confronto tra l'algoritmo FastICA e TCA. Il metodo di stima della contrast function usato è quello KDE.

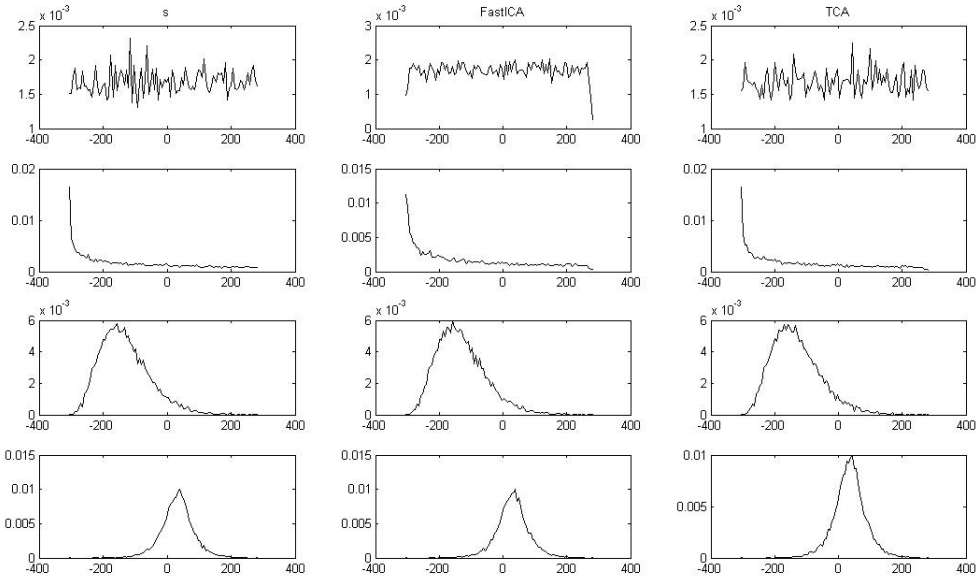


Figura 4.3: Nella prima colonna sono rappresentate le componenti del vettore s, Nella seconda e nella terza colonna sono rappresentate le componenti stimate con l'algoritmo FastICA e TCA rispettivamente.

La divergenza di Amari risulta:

- $\mathcal{D} = 2.4397$ con l'algoritmo FastICA
- $\mathcal{D} = 2.3712$ con l'algoritmo TCA

Si osserva un lieve miglioramento nel caso del TCA.

Secondo caso

Nel secondo caso abbiamo generato i dati in accordo con l'albero in figura 4.4

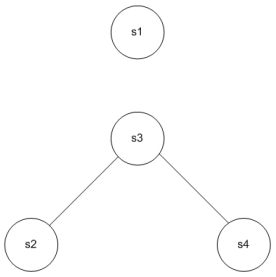


Figura 4.4: Modello dell'albero su cui fattorizzano i dati sintetici

La lista delle variabili aleatorie usate è:

- $u \sim \mathcal{U}$ una variabile aleatoria uniformemente distribuita
- $g \sim \Gamma$ una variabile aleatoria con distribuzione gamma
- $l \sim \mathcal{L}$ una variabile aleatoria di Laplace
- $n \sim \mathcal{N}$ una variabile aleatoria gaussiana

Da queste abbiamo costruito le componenti del vettore s nel seguente modo:

- $s_1 = u$
- $s_2 = g$
- $s_3 = g^2 + l$
- $s_4 = g \cdot n$

Nella figura 4.5 presentiamo il confronto tra il metodo KDE e quello KGV.

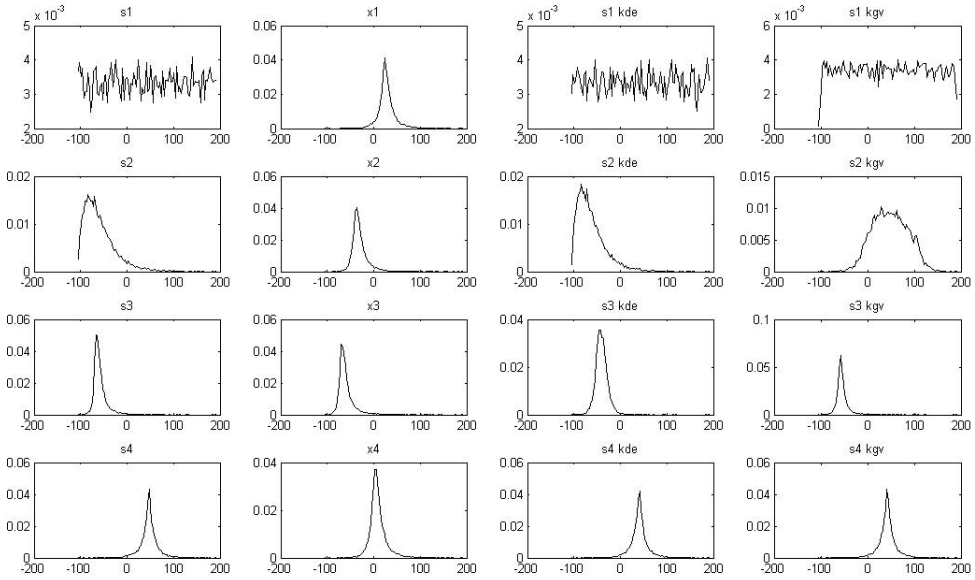


Figura 4.5: Nella prima colonna sono rappresentate le componenti del vettore s, la seconda colonna rappresenta le componenti del vettore dei dati x. Nelle ultime due colonne sono rappresentate le componenti stimate di s con i due metodi (KDE nella terza colonna e KGV nella quarta).

La divergenza di Amari risulta:

- $\mathcal{D} = 1.2961$ per il metodo KDE
- $\mathcal{D} = 2.3701$ per il metodo KGV

Come nel primo caso, anche questa volta il metodo KDE risulta migliore. Entrambi i metodi riescono a stimare bene l'albero.

Presentiamo ora il confronto (figura 4.6) tra l'algoritmo FastICA e TCA. Per il TCA useremo il metodo KDE.

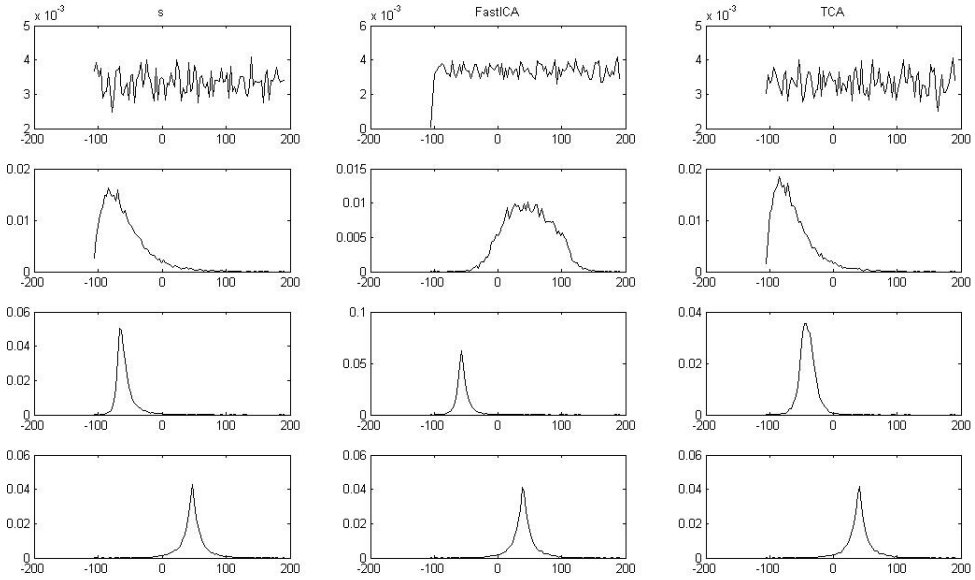


Figura 4.6: Nella prima colonna sono rappresentate le componenti del vettore s, Nella seconda e nella terza colonna sono rappresentate le componenti stimate con l'algoritmo FastICA e TCA rispettivamente.

La divergenza di Amari risulta:

- $\mathcal{D} = 2.1896$ con l'algoritmo FastICA
- $\mathcal{D} = 1.2961$ con l'algoritmo TCA

Si osserva un miglioramento nel caso del TCA.

4.3 Dati astrofisici sintetici

In questa sezione presenteremo i risultati ottenuti con dei dati simulati in accordo con le conoscenze teoriche sulle sorgenti astrofisiche. Le sorgenti che utilizzeremo sono il CMB, la free-free emission, il galactic dust e il synchrotron. Prenderemo in considerazione i canali a 30, 44, 70, 100 GHz. Da studi teorici è noto che il CMB è indipendente dalle altre sorgenti mentre la free-free emission, il dust e il synchrotron sono dipendenti una dall'altra.

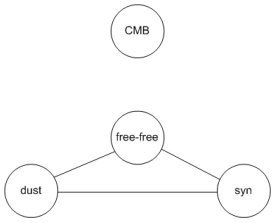


Figura 4.7: Struttura di dipendenza delle sorgenti astrofisiche

Come abbiamo visto, questa struttura di dipendenza non soddisfa le ipotesi dell'algoritmo TCA in quanto il grafo in figura 4.7 non è un albero. Applicando il TCA ai dati astrofisici avremo un'approssimazione della soluzione.

Useremo due patch di dati: una fuori dal piano galattico e una relativa al piano galattico. I dati di questi patch sono molto diversi tra loro. Fuori dal piano galattico infatti il CMB è la sorgente dominante, le altre sono di circa due o tre ordini di grandezza più piccole. Questo significa che, almeno nel range di frequenze utilizzate in queste simulazioni, il CMB è praticamente l'unica sorgente presente. Nel piano galattico invece la componente dominante è ovviamente il dust (due o tre ordini di grandezza più grande delle altre). Questa netta predominanza del dust rende molto più difficile separare il CMB in questo tipo di dati. Vedremo però che con il TCA si riesce ad avere dei buoni rusultati in entrambe i casi.

Confronteremo le prestazioni del TCA prima con quelle dell'algoritmo ICA e poi con gli altri algoritmi di dependent component analysis descritti nel Capitolo 1, Multidimensional ICA, Topographic ICA e CorCA.

4.3.1 Confronto tra ICA e TCA

Partiamo con le simulazioni relative a porzioni di cielo esterne al piano galattico. Il pixel centrale di ogni patch si trova a 20° dal piano galattico.

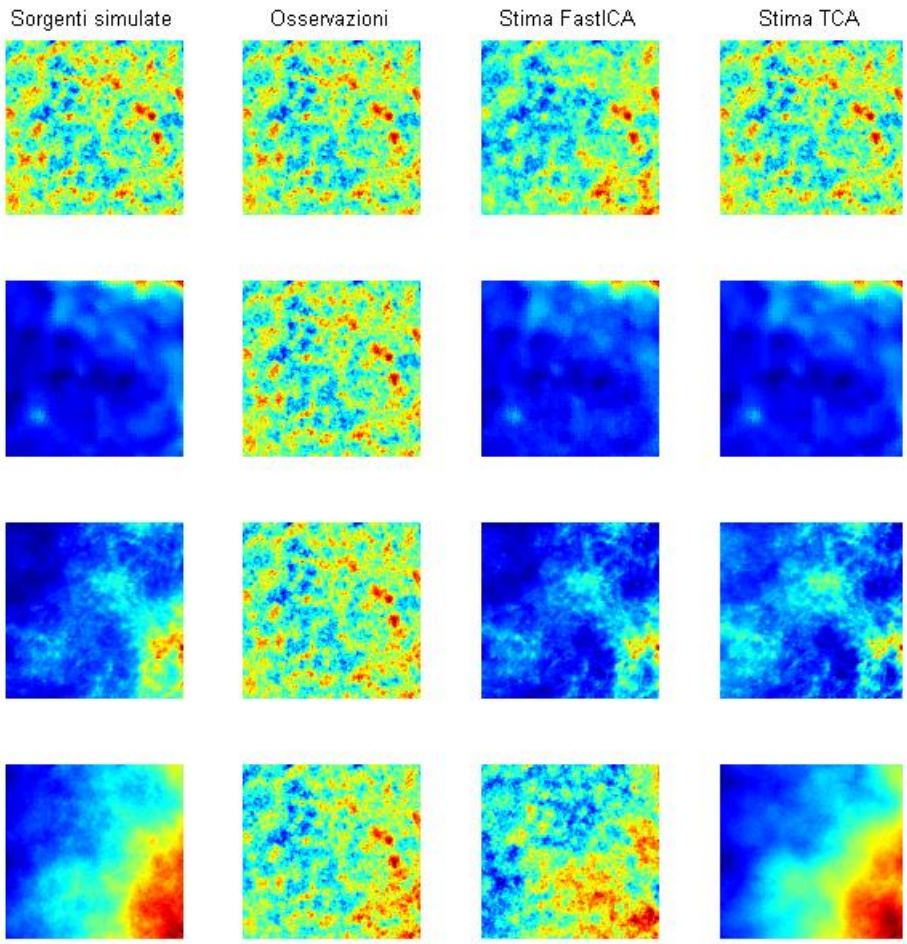


Figura 4.8: Nella prima colonna sono rappresentate le quattro sorgenti (dall'alto verso il basso) CMB, free-free emission, dust e synchrotron e nella seconda le osservazioni simulate sui quattro canali. Le altre due colonne rappresentano la stima di queste sorgenti prima con l'algoritmo FastICA e poi con il TCA

La divergenza di Amari risulta:

- $\mathcal{D} = 6.7795$ per l'algoritmo FastICA
- $\mathcal{D} = 3.4024$ per l'algoritmo TCA

Osservando la figura 4.8 possiamo verificare quanto detto in precendenza. Vediamo subito, guardando la colonna relativa alle osservazioni, che il CMB è la componente diminante. Le altre tre componenti, presenti in combinazione lineare con coefficienti diversi per ciascun canale, sono quasi assenti. Si può vedere poi che l'algoritmo FastICA non è riuscito a recuperare il synchrotron e presenta problemi anche nella stima del CMB. L'algoritmo TCA difetta leggermente nella stima del dust. Le prestazioni del TCA rimangono comunque globalmente superiori. Questo fatto è evidenziato dalla divergenza di Amari, più piccola nel caso TCA.

Occupiamoci ora della struttura di dipendenza.

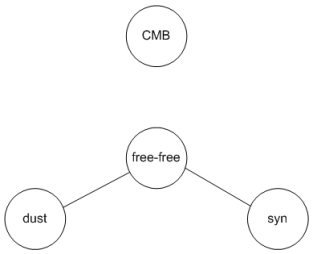


Figura 4.9: Struttura di dipendenza stimata dal TCA

Nella figura 4.9 possiamo vedere l'albero stimato dal TCA. Quest'ultimo è diverso da quello reale presentato in figura 4.7, ma risulta una migliore approssimazione rispetto all'algoritmo ICA.

Le informazioni mutue stimate (con la contrast function KDE) sono:

Informazioni mutue stimate	Dust	Free-free emission	Synchrotron	CMB
Dust	N.A.	0.0931	0.2864	0.0174
Free-free emission	0.0931	N.A.	0.2514	0.0391
Synchrotron	0.2864	0.2514	N.A.	0.0403
CMB	0.0174	0.0391	0.0403	N.A.

Tabella 4.1: Tabella delle informazioni mutue stimate per i dati fuori dal piano galattico

Passiamo ora alle simulazioni nel piano galattico. Nella figura 4.10 è presentato il confronto fra FastICA e TCA.

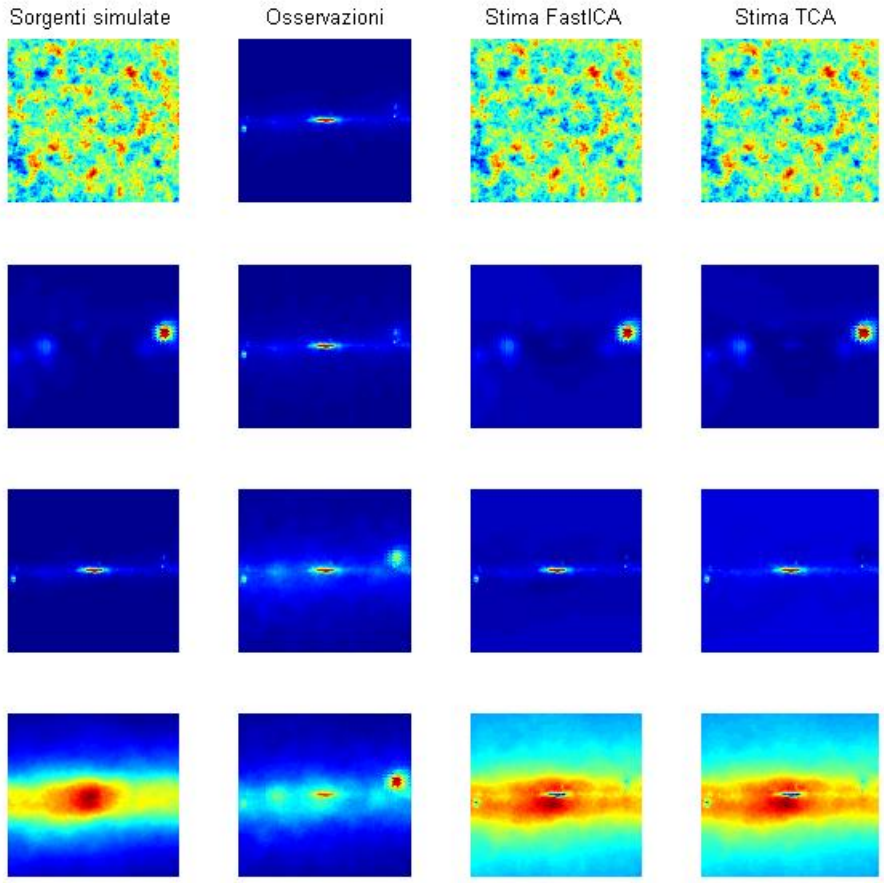


Figura 4.10: Sono rappresentate nella prima colonna le sorgenti(CMB, free-free emission, dust e synchrotron), nella seconda le osservazioni; le altre due colonne rappresentano la stima con l'algoritmo FastICA e con il TCA

Possiamo verificare subito dalle osservazioni che la componente dominante è quella del galactic dust. Con questi dati le prestazioni dell'algoritmo FastICA e del TCA sono molto simili tra loro. Entrambi ricostruiscono bene il CMB, la free-free emission e il dust, mentre nel synchrotron rimane pesante traccia del dust che non si è riusciti a eliminare.

Consideriamo ora la struttura di dipendenza. C'è da dire che la contrast function KDE ha avuto problemi a stimare la dipendenza tra le sorgenti. Questo è causato dal fatto che il dust, in questi dati, è molto forte e concentrato in una piccola porzione di pixel centrali. Questo porta ad avere un'istogramma quasi impulsivo, molto difficile da stimare con la tenica KDE. Il metodo KGV si è invece dimostrato più robusto sotto questo aspetto. La stima delle informazioni mutue è stata quindi ottenuta con il metodo KGV.

Informazioni mutue stimate	Dust	Free-free emission	Synchrotron	CMB
Dust	N.A.	0.1359	0.2040	0.0030
Free-free emission	0.1359	N.A.	0.1654	0.0029
Synchrotron	0.2040	0.1654	N.A.	0.0064
CMB	0.0030	0.0029	0.0064	N.A.

Tabella 4.2: Tabella delle informazioni mutue stimate per i dati dentro il piano galattico

4.3.2 Confronto tra Multidimensional ICA, Topographic ICA e TCA

Usando le stesse patch di dati, facciamo un confronto fra i vari algoritmi di dependent component analysis. Iniziamo con le patch fuori dal piano galattico.

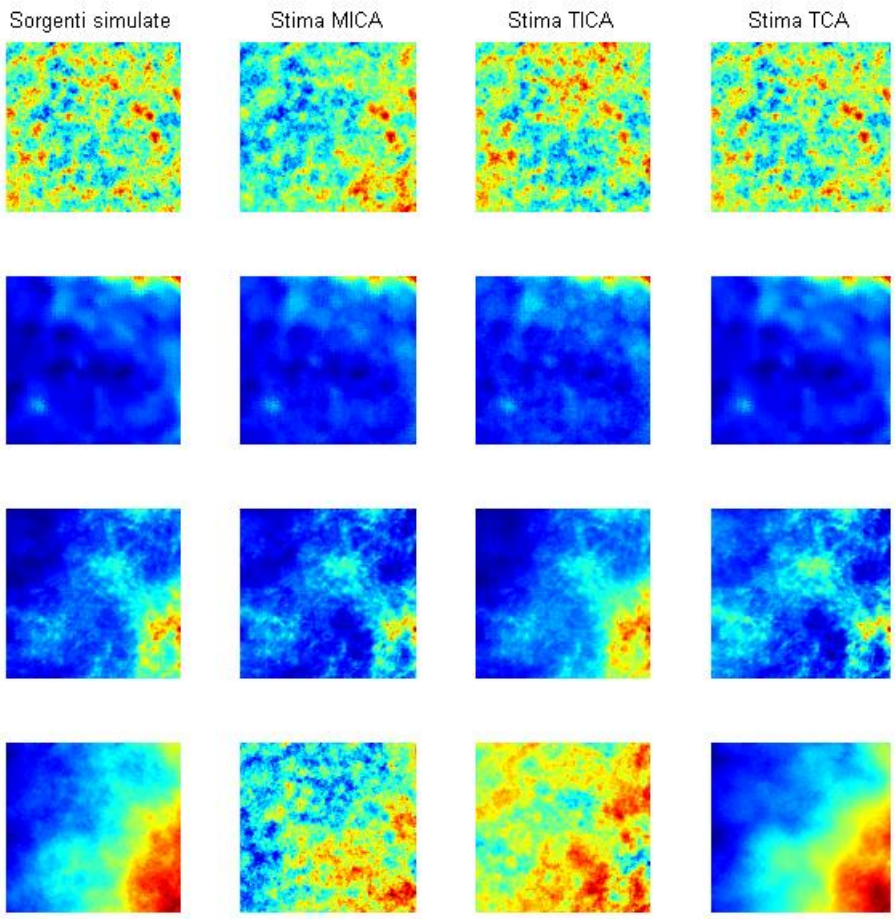


Figura 4.11: Sono rappresentate nella prima colonna le sorgenti(CMB, free-free emission, dust e synchrotron), nella seconda la stima con l'algoritmo Multidimensional ICA, nella terza la stima con l'algoritmo Topographic ICA e nell'ultima colonna i risultati TCA.

La divergenza di Amari risulta:

- $\mathcal{D} = 6.4167$ per l'algoritmo Multidimensional ICA
- $\mathcal{D} = 4.7032$ per l'algoritmo Topographic ICA
- $\mathcal{D} = 3.4024$ per l'algoritmo TCA

Come si può vedere sia dalla figura 4.11 sia dal valore della diver-

genza di Amari l'algoritmo TCA è quello che presenta prestazioni migliori.

Passiamo ora ai dati dentro il piano galattico.

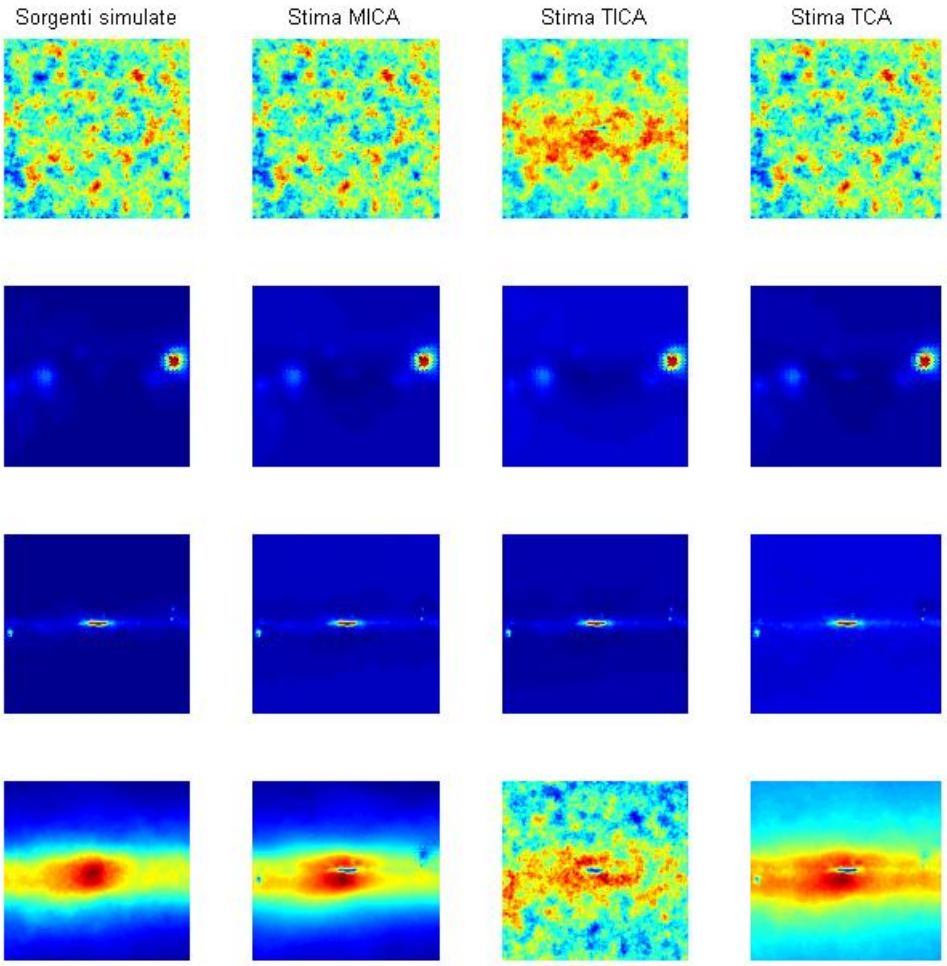


Figura 4.12: Sono rappresentate nella prima colonna le sorgenti(CMB, free-free emission, dust e synchrotron), nella seconda la stima con l'algoritmo Multidimensional ICA, nella terza la stima con l'algoritmo Topographic ICA e nell'ultima colonna i risultati TCA.

La divergenza di Amari risulta:

- $\mathcal{D} = 2.0185$ per l'algoritmo Multidimensional ICA
- $\mathcal{D} = 4.8760$ per l'algoritmo Topographic ICA
- $\mathcal{D} = 2.7618$ per l'algoritmo TCA

Possiamo notare un leggero miglioramento delle prestazioni dell'algoritmo MICA nei confronti del TCA. Come abbiamo detto, questo è dovuto al fatto che, in queste patch, il dust ha una distribuzione quasi impulsiva.

Notiamo anche le scarse prestazioni dell'algoritmo Topographic ICA. Questo significa che la dipendenza che lega le sorgenti astrofisiche non è del tipo descritto dalla (2.9). L'algoritmo TCA invece, non facendo nessuna particolare ipotesi sul tipo di dipendenza delle sorgenti, può trattare con qualsiasi tipo di dipendenza.

4.3.3 Confronto tra CorCA e TCA

Proponiamo ora un breve confronto tra gli algoritmi CorCA e TCA. Le patch utilizzate sono le stesse delle figure 4.8 e 4.10, avremo quindi come sorgenti il CMB, il dust, il synchrotron e la free-free emission. Anche i canali sono gli stessi: 30, 44, 70 e 100 GHz. Ricordiamo che l'algoritmo CorCA utilizza delle informazioni a priori sia sulla mixing matrix A, sia sulla struttura di correlazione. In particolare, nel nostro caso, conosciamo esattamente le colonne di A relative al CMB e alla free-free emission. In più sappiamo che la prima riga della matrice A è composta da tutti 1 a causa della normalizzazione. In sintesi, dei 16 coefficienti della mixing matrix (che nel nostro caso ha dimensione 4×4), dovremo stimare solo 6 coefficienti, quelli cioè delle colonne relative al dust e al synchrotron diversi da 1. Abbiamo inoltre l'informazione relativa alla matrice di correlazione. Come abbiamo già detto, il CMB risulta indipendente da tutte le altre sorgenti e quindi anche incorrelato. La matrice di correlazione delle sorgenti sarà quindi:

$$C_s(0,0) = \begin{pmatrix} \sigma_{11} & 0 & 0 & 0 \\ 0 & \sigma_{22} & \sigma_{23} & \sigma_{24} \\ 0 & \sigma_{32} & \sigma_{33} & \sigma_{34} \\ 0 & \sigma_{42} & \sigma_{43} & \sigma_{44} \end{pmatrix} \tag{4.2}$$

Ricordiamo che l'algoritmo TCA non utilizza nessuna informazione a priori. Questo lo rende molto più flessibile in quanto può essere utilizzato con successo in applicazioni in cui non si hanno conoscenze a priori sul fenomeno analizzato.

La divergenza di Amari per le patch relative a porzioni di cielo fuori dal piano galattico (figura 4.13) risulta risulta:

- $\mathcal{D} = 2.06324$ per l'algoritmo CorCA
- $\mathcal{D} = 3.4024$ per l'algoritmo TCA

mentre per le patch nel piano galattico (figura 4.14) avremo:

- $\mathcal{D} = 1.6458$ per l'algoritmo CorCA
- $\mathcal{D} = 2.7618$ per l'algoritmo TCA

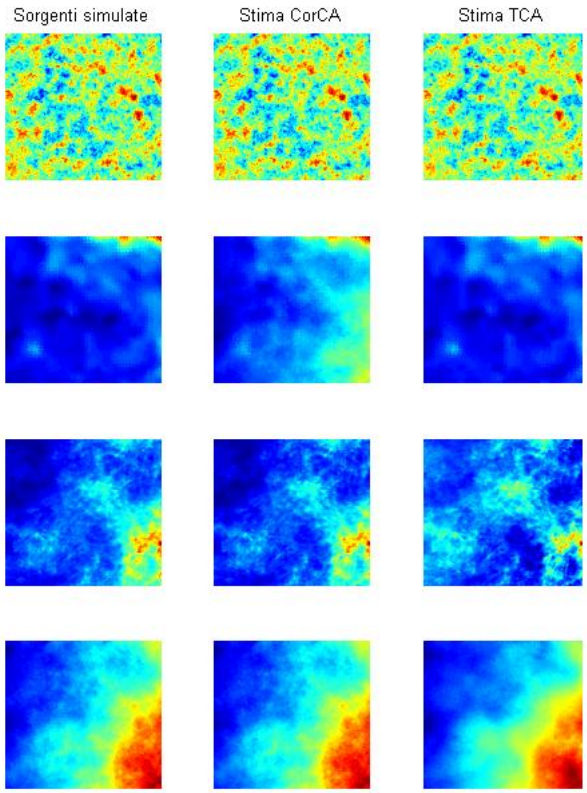


Figura 4.13: Porzione di cielo fuori dal piano galattico. Sono rappresentate nella prima colonna le sorgenti(CMB, free-free emission, dust e synchrotron), le altre due colonne rappresentano la stima con l'algoritmo CorCA e con il TCA

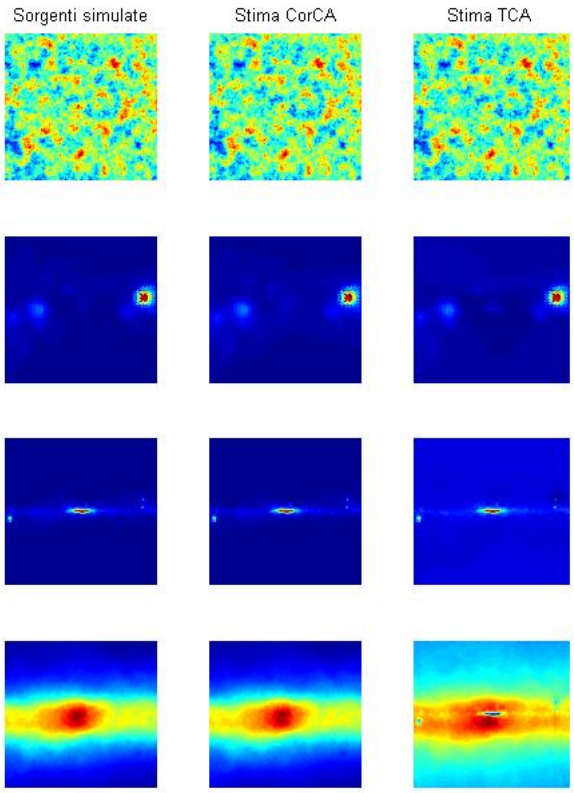


Figura 4.14: Porzione di cielo nel piano galattico. Sono rappresentate nella prima colonna le sorgenti(CMB, free-free emission, dust e synchrotron), le altre due colonne rappresentano la stima con l'algoritmo CorCA e con il TCA

C'è da dire però che l'algoritmo CorCA è fortemente dipendente dall'informazione a priori. Se, ad esempio, togliessimo i vincoli di incorrelazione tra il CMB e le altre sorgenti le prestazioni risulterebbero di molto degradate (figure 4.15 e 4.16).

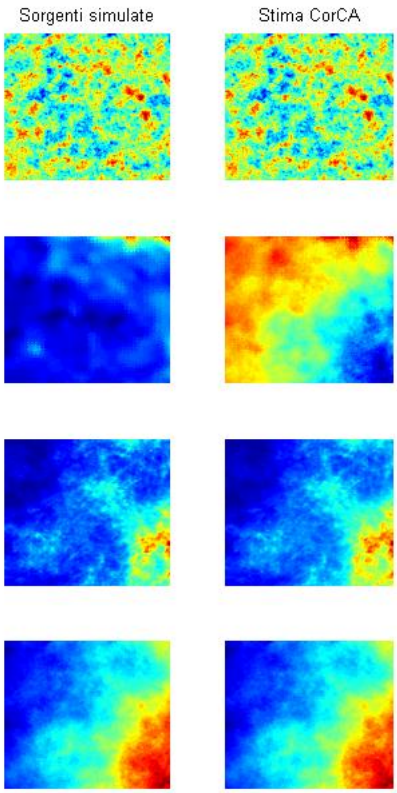


Figura 4.15: Stima senza il vicolo dell'incorrelazione per la porzione di cielo fuori dal piano galattico. Sono rappresentate nella prima colonna le sorgenti(CMB, free-free emission, dust e synchrotron), nella seconda la stima con l'algoritmo CorCA

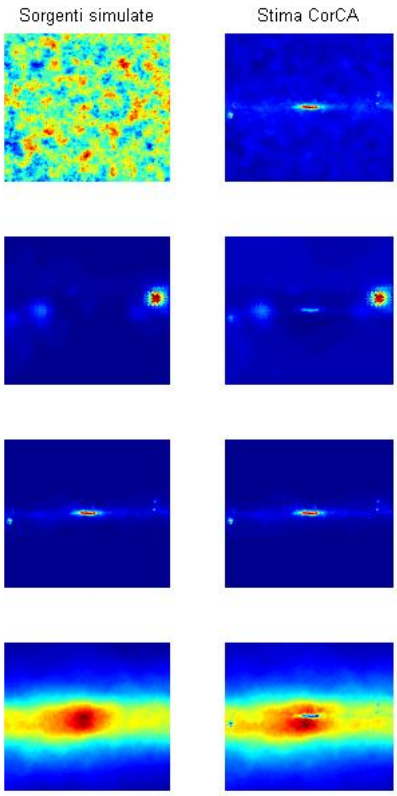


Figura 4.16: Stima senza il vicolo dell'incorrelazione per la porzione di cielo nel piano galattico.Sono rappresentate nella prima colonna le sorgenti(CMB, free-free emission, dust e synchrotron), nella seconda la stima con l'algoritmo CorCA

4.3.4 Problema del rumore

In questa sezione parleremo del problema del rumore generato dagli strumenti. Tutte le strumentazioni elettroniche producono un rumore termico dovuto alla loro temperatura. Come abbiamo già detto i rivelatori del satellite Planck sono fortemente raffreddati (fino a 20 K quelli a bassa frequenza e a 4 K quelli ad alta frequenza). I segnali con cui dobbiamo lavorare però sono molto deboli e spesso il livello di rumore supera di molto il livello del

segnale utile. E' possibile seguire diversi approcci al problema del rumore che possono essere riassunti in due classi:

- pre-filtraggio
- filtraggio integrato nell'algoritmo

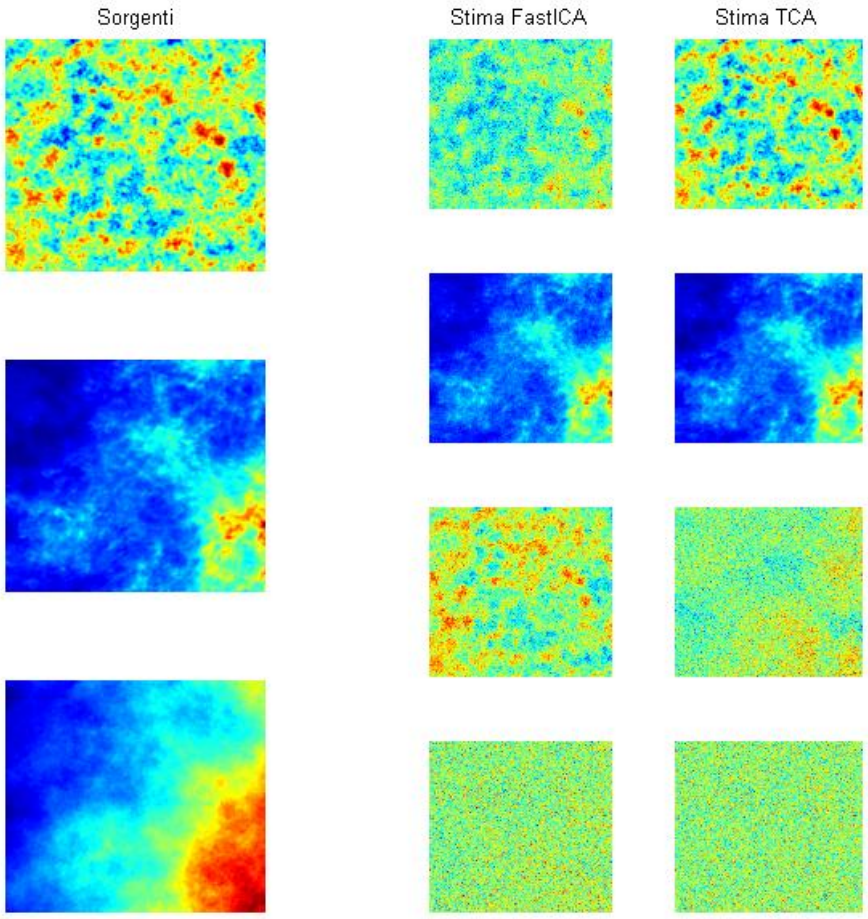
Nelle simulazioni assumeremo il rumore gaussiano e stazionario. La deviazione standard per ogni canale frequenziale (nell'ordine da 30 a 857 GHz) sarà: 0.00126, 0.00120, 0.00113, 0.00028, 0.00018, 0.00018, 0.00018, 0.00018, 0.00018. Ricordiamo che questo modello di rumore è solo un'approssimazione, in quanto in realtà è fortemente non-stazionario.

Un primo metodo molto semplice per eliminare il rumore si basa sulla caratteristica dell'algoritmo TCA di essere "blind", di non avere cioè nessuna informazione a priori sulle sorgenti. Partiamo richiamando il modello delle osservazioni:

$$x = As + n \tag{4.3}$$

con $x, n \in \mathbb{R}^m$ dove m indica il numero dei canali, $s \in \mathbb{R}^n$ con n pari al numero delle sorgenti e A la mixing matrix di dimensioni $m \times n$ con $m \geq n$. Supponiamo ora che le sorgenti siano in numero strettamente minore dei canali. Un algoritmo blind riceverà quindi in ingresso un vettore m-dimensionale, e non avendo nessuna informazione sul numero delle sorgenti, stimerà una demixing matrix W di dimensioni $m \times m$. Di conseguenza il vettore delle sorgenti stimate $\hat{s} = Wx$ sarà anch'esso m-dimensionale. Di queste m componenti n saranno le stime delle sorgenti originali, $m - n$ invece saranno combinazioni lineari delle componenti di rumore.

Per provare quanto appena detto presentiamo la seguente simulazione: abbiamo simulato un sistema con quattro canali frequanziali (30, 44, 70 e 100 GHz) prendendo solo tre sorgenti (CMB, dust e synchrotron). La mixing matrix A avrà dimensione 4×3. Abbiamo poi aggiunto il vettore di rumore con quattro componenti gaussiane con la deviazione standard propria di ciascun canale. I risultati ottenuti sono presentati nella figura 4.17.



(a) (b)

Figura 4.17: (a) Le tre sorgenti utilizzate (CMB, dust, synchrotron) (b) Confronto fra la stima ottenuta con l'algoritmo FastICA e quella del TCA. Come si può vedere dalle prime due immagini, il rumore è stato ben separato dal CMB e dal dust. Questo non è avvenuto per il synchrotron, che è di gran lunga la sorgente più debole.

Osservando la figura 4.17(b) si nota subito che il synchrotron non è stato ricostruito nè dall'ICA nè dal TCA. Il motivo è semplice: il rumore degli strumenti per questi canali frequenziali ha una varianza di circa il 30% rispetto a quella del CMB. Ora il synchrotron è di ben 3 ordini di grandezza più debole del CMB e

di conseguenza almeno 2 ordini di grandezza più debole del rumore. A parte questo però notiamo che l'algoritmo TCA riesce a separare le altre due sorgenti dal rumore molto meglio dell'algoritmo ICA. Inoltre il dust è stato ricostruito in maniera più precisa di quanto non si era riusciti a fare nelle simulazioni senza rumore (figura 4.11). Questo perchè nelle simulazioni precedenti avevamo quattro canali per quattro sorgenti; qui invece usiamo quattro canali per tre sorgenti. Abbiamo quindi a disposizione un maggior numero di informazioni. Ricordiamo infine che nel satellite Planck ci sono ben nove canali frequanziali. Questo assicura una grande quantità di informazioni da usare nella separazione, una quantità molto maggiore rispetto a quella usata nelle nostre simulazioni.

Pre-filtraggio

Se sfruttare l'informazione proveniente da tutti i canali può risultare efficacie per separare le sorgenti più forti dal rumore; questa strategia non funziona con le sorgenti più deboli (ad esempio il synchrotron). Dato il modello delle osservazioni (4.3) si può subito pensare di eliminare il rumore prima di passare i dati all'algoritmo di separazione. Ci sono molti metodi per fare questa operazione. Noi, in questa tesi, ne abbiamo usati due tra i più semplici: il filtro di Wiener e il filtro di Kalman. In particolare abbiamo filtrato ogni componente del vettore delle osservazioni, passando poi i dati all'algoritmo di separazione. I risultati però sono stati insoddisfacenti. La principale causa di questo sta nel fatto che il nostro problema non soddisfa le ipotesi necessarie per il funzionamento dei due filtri. Sia il rumore sia i dati astrofisici sono fortemente non stazionari. L'ipotesi di stazionarietà sia dei dati sia del rumore è invece fondamentale per l'applicabilità del filtro di Wiener, mentre per il filtro di Kalman deve essere stazionario solo il rumore. Diciamo però che ci sono metodi più avanzati che tengono in considerazione la struttura spazio-variante sia delle sorgenti che del rumore (ad esempio il *particle filter*). Lasceremo a studi futuri ulteriori sviluppi di questi metodi.

Filtraggio integrato nell'algoritmo

Discutiamo ora un metodo di filtraggio interno all'algoritmo. Le ipotesi che assumeremo sono la stazionarietà e la gaussianità dei dati di rumore e la conoscenza della matrice di covarianza del rumore stesso. L'idea è quella di fare una stima a massima verosimiglianza del vettore delle osservazioni s al passo k e ripetere questa stima per ogni k. Vediamo più in dettaglio questo metodo.

Nello schema 4.18 è mostrato il funzionamento del TCA in assenza di rumore.

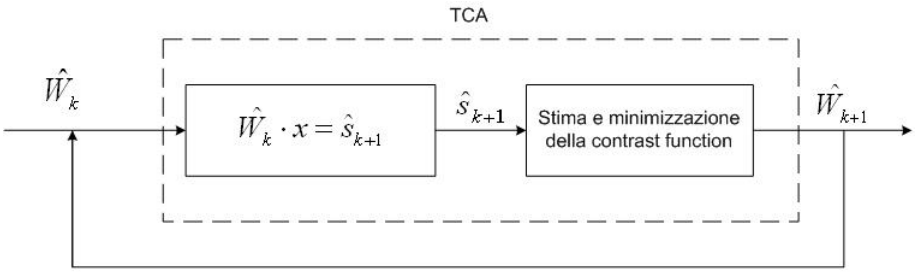


Figura 4.18: Schema di funzionamento del TCA.

Questo schema suggerisce immediatamente una possibile estensione al caso di dati rumorosi. Basta infatti sostituire nel primo blocco la stima a massima verosimiglianza (ML) di $\hat{s}_{k+1}$.

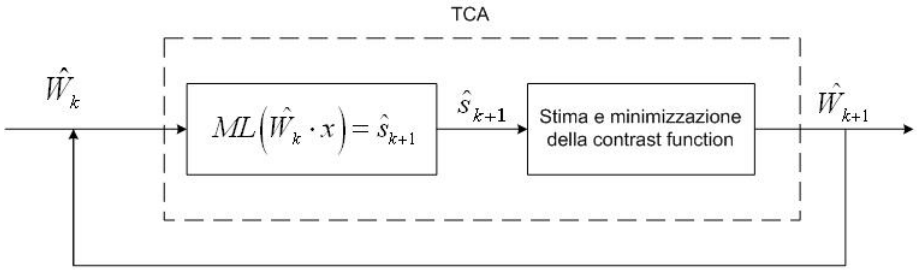


Figura 4.19: Schema di funzionamento del TCA con rumore.

Vediamo ora come fare la stima ML. Per ogni pixel j vale:

$$x_j = As_j + n_j$$

dove n_j è un vettore gaussiano a media nulla e con matrice di covarianza Σ. La sua densità di probabilità può quindi essere scritta come:

$$p_{n_j}(n_j) = (2\pi)^{-n/2} |\Sigma|^{-1/2} \exp\left(-\frac{1}{2} n_j^t \Sigma^{-1} n_j\right)$$

Da qui in poi indicheremo con T l'inversa di Σ. Assumendo ora tutti i pixel indipendenti possiamo scrivere la ddp dell'insieme di dati che stiamo considerando come:

$$p_x(x\,|A, s, T) = \prod_{j=1}^{N} (2\pi)^{-n/2} |T|^{-1/2} \exp\left(-\frac{1}{2}(x_j - As_j)^t T (x_j - As_j)\right) =$$

$$= (2\pi)^{-n/2} |T|^{-1/2} \exp\left(-\frac{1}{2}\sum_{j=1}^{N}(x_j - As_j)^t T (x_j - As_j)\right)$$

Come di consueto definiamo la funzione di verosimiglianza di s nel solito modo:

$$L(s) = (2\pi)^{-n/2} |T|^{-1/2} \exp\left(-\frac{1}{2}\sum_{j=1}^{N}(x_j - As_j)^t T (x_j - As_j)\right)$$

Prendiamo ora il logaritmo della funzione di verosimiglianza e scartiamo i termini costanti che non dipendono da s:

$$\ln L(s) = -\frac{1}{2}\sum_{j=1}^{N}(x_j - As_j)^t T (x_j - As_j)$$

Indichiamo con $F(s_j)$ la forma quadratica $(x_j - As_j)^t T (x_j - As_j)$. L'ultimo passo sarà quello di minimizzare rispetto a s_j la funzione di verosimiglianza come imposto dalla tecnica ML. In formule:

$$\frac{d}{ds_j}\ln L(s) = -\frac{1}{2}\sum_{j=1}^{N}\frac{d}{ds_j}F(s_j) = 0$$

Siccome $F(s_j) \geq 0$ per qualsiasi j, basterà risolvere $\frac{d}{ds_j}F(s_j) = 0$ per qualsiasi j. Per non appesantire troppo la trattazione, tralasciamo i passaggi matematici (consultare l'appendice C) e diamo il risultato finale:

$$s = A^{-1}x \tag{4.4}$$

dove s e x vanno intesi come due matrici che contengono s_j e x_j come colonne.

Come si può vedere, purtroppo nel caso di rumore gaussiano a media nulla la stima ML non ci aiuta. L'equazione (4.4) non dipende dal livello di rumore, se quindi i dati sono molto rumorosi la stima risulterà pessima. Si può pensare di arricchire questo metodo con eventuali informazioni a priori sulla densità di probabilità delle sorgenti. Con questa eventuale informazione si può pensare di passare al criterio di stima MAP (maximum a posteriori probability). Ci sono poi altri problemi: per risolvere l'equazione (4.4) infatti dobbiamo conoscere la mixing matrix A. Ad ogni passo k questa è ottenuta invertendo la demixing matrix $\hat{W}_k$ stimata. A parte l'invarianza alla permutazione che non comporta peggioramenti della stima, l'invarianza a fattori di scala e alla combinazione lineare di nodo foglia e genitore possono essere potenzialmente molto dannose. Scalare le componenti, ad esempio, comporta una variazione in nessun modo prevedibile della potenza del segnale utile rispetto a quella di rumore.

4.4 Dati WMAP

In questa sezione presenteremo i risultati ottenuti con dati WMAP. Abbiamo parlato in precedenza di questa missione, entriamo ora in una descrizione più precisa dei dati.

I dati WMAP provengono da cinque canali frequenziali (K, Ka, Q, V, W) con frequenza di centro banda pari a: 22, 30, 40, 60, 94 GHz. Come possiamo osservare dalla figura 3.2, sulla banda dei canali K e Ka è predominante il synchrotron mentre nella banda del canale W la sorgente dominante è il CMB. Osserviamo invece che il dust, alle frequenze del WMAP, è molto al di sotto delle altre sorgenti.

Prima di poter elaborare questo tipo di dati bisogna risolvere alcuni problemi di natura geometrica. Il satellite WMAP osserva l'intera volta celeste, questo significa che i dati sono presi da una superficie sferica. Ora per convertire questi dati in un'immagine bisogna definire pei pixel. E' noto però che è impossibile suddividere una superficie sferica in pixel quadrati. Bisogna quindi suddividere la sfera in pixel con una forma che varia in

funzione della loro posizione nella volta celeste. Ci sono molti metodi per fare questa suddivisione. Per i dati WMAP è stato usato lo schema HEALPix (Hierarchical Equal Area isoLatitude Pixelization scheme) (figura 4.20).

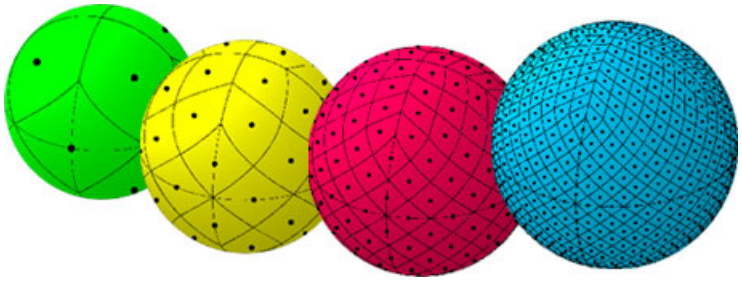


Figura 4.20: Pixellizzazione della sfera con lo schema HEALPix

C'è un altro problema di natura geometrica: per l'elaborazione finale dei dati servono delle patches piane. Dopo aver suddiviso la sfera in pixel, bisogna sceglierne una porzione e proiettarla sul piano tangente alla sfera. La grandezza di una patch è un parametro critico: se infatti scegliessimo una porzione di sfera troppo grande, la proiezione presenterà forti distorsioni. La grandezza ottimale per dati WMAP risulta essere di 128×128. C'è un ultimo problema da affrontare che riguarda la risoluzione angolare. Per ogni canale frequenziale, il telescopio ha una diversa configurazione ottica. In altre parole, ogni canale ha una diversa *Point Spread Function*. I dati presentano quindi una risoluzione (e una aberrazione) che varia da canale a canale. Questo complica di molto il problema della separazione delle componenti. Il modello lineare che abbiamo assunto ($x = As$) infatti non è più valido, bisognerebbe usare un modello convolutivo. Per evitare questo problema, i dati WMAP che useremo sono stati "degradati" per avere una risoluzione di 1° comune a tutti i canali. La dimensione dei pixel viene scelta più piccola della risoluzione angolare in modo da avere un sovracampionamento. Nei dati WMAP che utilizzeremo la dimensione del pixel è la stessa per ogni canale e vale 6.87 arcmin (circa nove volte più piccola della risoluzione angolare). A questo punto valgono tutte le ipotesi di applicabilità degli algoritmi di separazione.

Passiamo ora ad analizzare i risultati ottenuti. Ricordiamo che i dati WMAP sono dati reali (non simulazioni), per questo valutare l'accuratezza della stima può risultare difficile. Non sappiamo

infatti come sono le sorgenti reali, non abbiamo quindi nulla con cui confrontare i nostri risultati. In questa tesi prenderemo come riferimento i risultati ottenuti dal gruppo WMAP. Cerchiamo di capire cosa dovremmo aspettarci. Sulla bande usate dal WMAP i componenti più rilevanti sono il synchrotron e il CMB. La free-free emission si fa sentire solo su una piccola finestra intorno ai 70 GHz, alle altre frequenze è sempre dominata dal synchrotron o dal CMB. Il dust è presente soprattutto nelle bande V e W ma è comunque molto più debole rispetto al CMB.

4.4.1 Patch 1

Questa patch riguarda una porzione di cielo situata leggermente sopra il piano Galattico. Le coordinate esatte del pixel centrale sono: latitudine Galattica 20° nord e longitudine Galattica 60° ovest. In questa posizione dovremmo osservare, oltre al CMB che è una sorgente diffusa, qualche componente galattico (soprattutto synchrotron). Faremo un confronto fra i nostri risultati e quelli del gruppo WMAP. Facciamo notare che tutte le immagini, ad eccezione di quelle relative al CMB, ottenute dal gruppo WMAP hanno una minore risoluzione. Notiamo che nelle immagini relative al synchrotron e alla free-free emission (figure 4.21 e 4.22) ci sono degli elementi in comune: si distinguono infatti due sorgenti molto compatte e di forte intensità nella stessa posizione. Queste emissioni anomale sono dovute alle altre galassie vicine alla nostra e vengono dette “extragalactic point sources”. Per la loro natura quasi puntiforme, sono molto difficili da separare con un algoritmo di separazione. Il problema può però essere agevolmente risolto eliminando in anticipo i pixel relativi a questo tipo di sorgenti, dato che la loro posizione è nota. La stime del dust è particolarmente difficile perchè, come abbiamo detto, nelle bande del satellite WMAP questa sorgente emette molto debolmente. La cosa sarà molto diversa con il satellite Planck che avrà ben cinque canali ad alta frequenza.

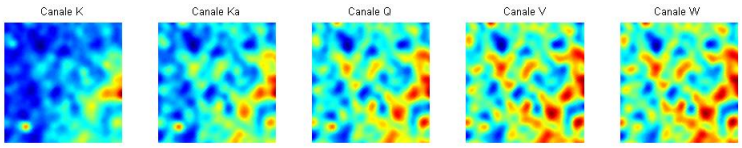
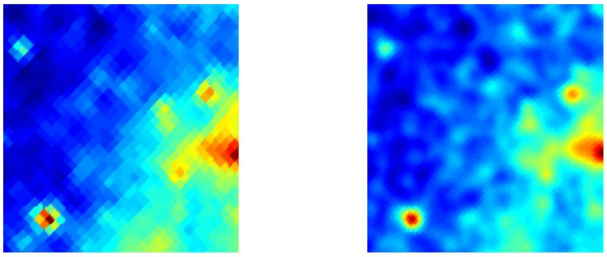
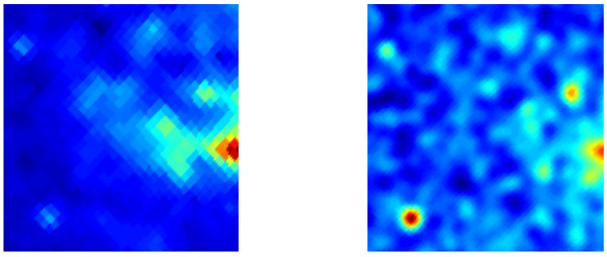


Figura 4.21: Dati provenienti da ciascuno dei cinque canali del WMAP.



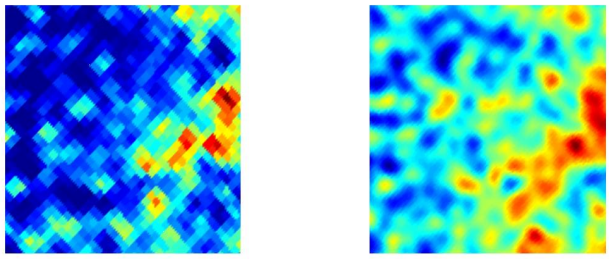
(a) (b)

Figura 4.22: Synchrotron stimato (a)dal gruppo WMAP, (b)con l'algoritmo TCA



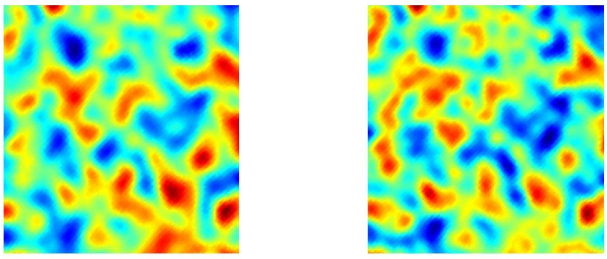
(a) (b)

Figura 4.23: Free-free emission stimata (a)dal gruppo WMAP, (b)con l'algoritmo TCA



(a) (b)

Figura 4.24: Dust stimato (a)dal gruppo WMAP, (b)con l'algoritmo TCA



(a) (b)

Figura 4.25: CMB stimato (a)dal gruppo WMAP, (b)con l'algoritmo TCA

4.4.2 Patch 2

La seconda patch è relativa a una porzione di cielo alla stessa latitudine dell'altra, ma con 90° ovest in più di longitudine. Le coordinate esatte del pixel centrale risultano essere: latitudine galattica 20° nord, longitudine galattica 150° ovest.

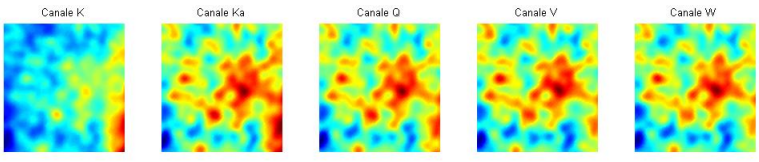
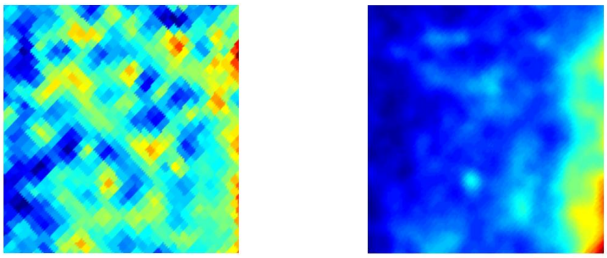
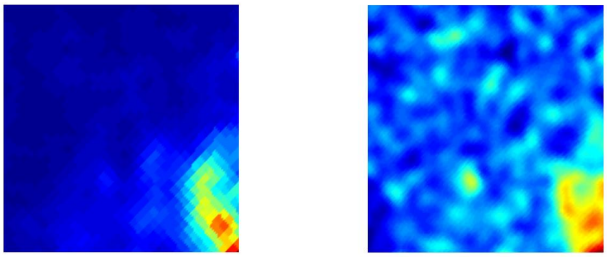


Figura 4.26: Dati provenienti da ciascuno dei cinque canali del WMAP.



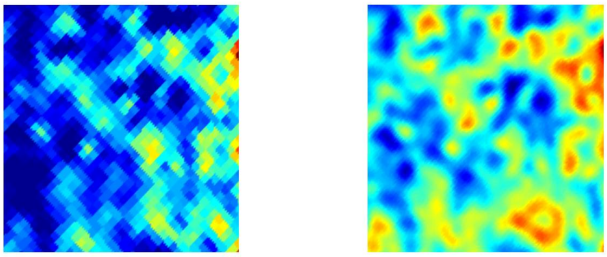
(a) (b)

Figura 4.27: Synchrotron stimato (a)dal gruppo WMAP, (b)con l'algoritmo TCA



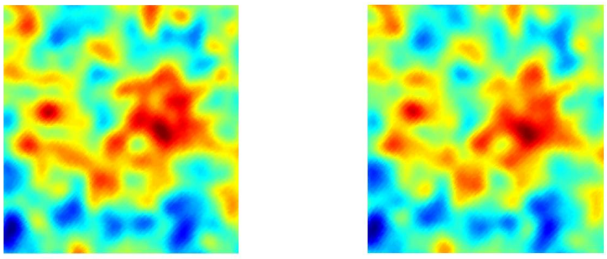
(a) (b)

Figura 4.28: Free-free emission stimata (a)dal gruppo WMAP, (b)con l'algoritmo TCA



(a) (b)

Figura 4.29: Dust stimato (a)dal gruppo WMAP, (b)con l'algoritmo TCA



(a) (b)

Figura 4.30: CMB stimato (a)dal gruppo WMAP, (b)con l'algoritmo TCA

Conclusioni

In questa tesi abbiamo discusso un algoritmo, il TCA, che mira ad estendere il classico approccio ICA a una gamma di problemi in cui la dipendenza tra le variabili non può essere trascurata. Abbiamo provato il TCA su dati sintetici sia generati in Matlab sia astrofisici. Da queste simulazioni è emerso che il TCA presenta migliori prestazioni non solo rispetto all'algoritmo ICA ma anche rispetto ad altri algoritmi di dependent component analysis quali Multidimensional ICA e Topographic ICA. Ci sono però degli aspetti che si prestano a successivi sviluppi:

- Ammettere strutture di dipendenza più generali degli alberi. L'algoritmo TCA ammette un numero di componenti totalmente connesse minore o uguale a 2 (figura 2.1). Il passo successivo sarà quello di estendere l'algoritmo a problemi con un numero qualsiasi di componenti connesse in modo da poter considerare ogni possibile ddp per il vettore delle sorgenti s.
- Problema del rumore. In questa tesi abbiamo solo accennato al problema del rumore proponendo metodi di pre-filtraggio e di filtraggio interno all'algoritmo. I metodi usati non hanno però dato i risultati sperati. Studi successivi dovranno mirare a sfruttare eventuali informazioni a priori. Nel problema astrofisico, ad esempio, il rumore è perfettamente noto in quanto prodotto dall'antenna del satellite. Una tecnica promettente per il filtraggio in presenza di segnale e rumore entrambi non stazionari è il *particle filter*.
- Sviluppo della contrast function KGV. Questo tipo di contrast function è uno strumento molto valido per stimare l'informazione mutua. Ci sono però alcuni problemi matematici che devono ancora essere risolti ([28], [20]): il primo riguarda il doppio passaggio al limite ($\Delta x, \Delta y \to 0$ e $\sigma \to 0$)

nelle relazioni (2.101, 2.102); il secondo riguarda il fatto che la proiezione di s in un RKHS non conserva l'indipendenza condizionata; infine la dimostrazione fatta nel caso particolare di due variabili aleatorie deve essere estesa al caso generale di m vettori aleatori.

Ringraziamenti

Alla fine di questa tesi, ma soprattutto alla fine di sei lunghi anni di università, devo necessariamente fare dei ringraziamenti. In primo luogo ringrazio la mia famiglia, i miei genitori Manfredo e Maria Pia e le mie due sorelle Claudia e Federica, per il sostegno economico e morale. Senza di loro non sarei mai arrivato a scrivere questa tesi. Ringrazio tutti i miei amici e compagni di corso Gianni, Paolo, Christian, Anthony, Simone, Angela, Beatrice, Andrea (il Dippi), Alberto e tutti gli altri che qui non ho citato ma che meritano tutta la mia riconoscenza (sei anni di amicizie non si possono racchiudere in poche righe). Ringrazio infine i miei relatori, il professor Ercan Kuruoglu e la professoressa Maria Sabrina Greco che mi hanno aiutato e seguito in questo lavoro finale.

Bibliografia

[1] A.Aissa-El-Bey, K.Abed-Meraim and Y. Grenier "Blind separation of audio source convolutive mixtures using parametric decomposition" *IEEE Transactions on Audio, Speech and Language, July, 2007.*

[2] E. Vincent, R. Gribonval and C. Févotte "Performance Measurement in Blind Audio Source Separation" *IEEE Transactions on audio, speech and language processing, Vol 14, No 4 July 2006.*

[3] E. Vincent, R. Gribonval and C. Févotte "A Tentative Typology of Audio Source Separation Tasks" *4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA 2003), April 2003, Nara, Japan .*

[4] S. Fiori "Overview of independent component analysis technique with an application to synthetic aperture radar (SAR) imagery processing" *Neural Networks 16(2003) 453-467.*

[5] J. Karhunen, A. Hyvärinen, R. Vigário, J. Hurri and E. Oja "Applications of Neural Blind Separation to Signal and Image Processing" *Proceedings of the 1997 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 1997).*

[6] F. H. Y. Chan, C. Chang, W. Xu, F. K. Lam and P. Kwok "Blind separation and Localization of Dipole Sources of MEG" *Proceedings of The First Joint BMES/EMBS Conference Serving Humanity, Advancing Technology 1999, Atlanta USA.*

[7] V. D. Calhoun, T. Adali, L. K. Hansen, J. Larsen and J. J. Pekar "ICA of Functional MRI Data: an Overview" *4th International Symposium on Independent Component Analysis*

and Blind Signal Separation (ICA 2003), April 2003, Nara, Japan.

[8] V. D. Calhoun, T. Adali, G. Pearlson "Independent component analysis applied to fMRI data: a generative model for validating results" *Neural Networks for Signal Processing XI, 2001. Proceedings of the 2001 IEEE Signal Processing Society Workshop.*

[9] C. A. Estombelo-Montesco, D. B. de Araujo "Dependet component analysis for the magnetogastrographic detection of human electrical response activity" *IOP Publishing Physiol. Meas. 28(2007) 1029-1044.*

[10] E. Oja, K. Kiviluoto and S. Malaroiu "Indipendent Component Analysis for Financial Time Series" *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. 1-4 Oct. 2000 Page(s):111 - 116* .

[11] Zhi-bin Lai, Yiu-ming Cheung and Lei Xu "Indipendent Component Ordering in ICA Analysis of Financial Data" http://citeseer.ist.psu.edu/lai99independent.html.

[12] A. Hyvärinen and E. Oja "Indipendent Component Analysis: Algorithms and Applications" *Neural Networks, 13(4-5):411-430, 2000.*

[13] P. Comon "Indipendent component analysis, A new concept?" *Signal Processing 36(1994) 287-314.*

[14] D. T. Pham "Blind Separation of Instantaneous Mixture of Sources via an Indipendent Component Analysis" *IEEE Transactions on signal processing, Vol 44, No 11, November 1996.*

[15] J.-F. Cardoso "Multidimensional Indipendent Component Analysis" *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 1998.*

[16] A. Hyvärinen, P. O. Hoyer and M. Inki "Topographic Indipendent Component Analysis" *Neural Computation 13(7):1527-1558 July 2001.*

[17] L. Bedini, D. Herranz, E. Salerno, C. Baccigalupi, E. E. Kuruoglu, A. Tonazzini "Separation of Correlated Astrophysical Sources Using Multiple-Lag Data Covariance Ma-

trices" *EURASIP Journal on Applied Signal Proccessing 2005:15,2400-2412.*

[18] F. R. Bach, M. I. Jordan "Beyond Independent Component: Tree and Clusters" *Journal of Machine Learning Research 4(2003) 1205-1233.*

[19] S. L. Lauritzen "Graphical Models" *Clarendon Press, 1996.*

[20] K. Fukumizu, F. R. Bach, M. I. Jordan "Dimensionality Reduction for Supervised Learning with Reproducing Kernel Hilbert Space" *Journal of Machine Learning Research 5 (2004) 73-99.*

[21] T. H. Cormen, C. E. Leiserson and R. L. Rivest "Introduction to Algorithms" *MIT Press, 1989.*

[22] B. W. Silverman "Density Estimation for Statistics and Data Analysis" *Chapman and Hall, 1985.*

[23] F. R. Bach, M. I. Jordan "Kernel independent component analysis" *Journal of Machine Learning Research, 3(2002)1-48.*

[24] S. Kullback "Information Theory and Statistics" *New York: John Wiley & Sons, 1959.*

[25] S. Saitoh "Theory of Reproducing Kernels and its Applications" *Harlow, UK: Longman Scientific & Technical, 1988.*

[26] F. Girosi, M. Jones and T. Poggio "Regularizaton theory and neural networks architectures" *Neural Computation, 7(2), 1995, 219-269.*

[27] K. Fukumizu, F. R. Bach and A. Gretton "Statistical Consistency of Kernel Canonical Correlation Analysis" *Journal of Machine Learning Research 8(2007) 361-383.*

[28] A. Gretton, R. Herbrich, A. Smola, O. Bousquet, B. Schölkopf "Kernel Methods for Measuring Independence" *Journal of Machine Learning Research 6(2005) 2075-2129.*

[29] T. M. Cover and J. A. Thomas "Element of Information Theory" *New York: John Wiley & Sons, 1991.*

[30] Sito della missione Planck: www.rssd.esa.int/Planck.

[31] R. A. Horn and C. R. Johnson "Matrix Analysis" *Cambridge University Press, Cambridge, 1985.*

[32] B. Schölkopf, A. J. Smola and K. R. Müller "Nonlinear component analysis as a kernel eigenvalue problem" *Neural Computation 10(3) 1299-1319, 1998.*

Appendice A

A.1 Divergenza di Kullback-Leibler e informazione mutua

Divergenza di Kullback-Leibler (DKL)

Siano p e q due densità di probabilità. La divergenza di Kullback-Leibler è definita come:

$$D\left(p \parallel q\right) = \int_{-\infty}^{\infty} p\left(x\right) \log \frac{p\left(x\right)}{q\left(x\right)} dx$$

La DKL non è una distanza nello spazio vettoriale delle ddp in quanto non è simmetrica ($D\left(p \parallel q\right) \neq D\left(q \parallel p\right)$) e non rispetta la disuguaglianza triangolare. E' però una buona misura di quanto distano due ddp dal punto di vista informativo. Valgono poi importanti proprietà:

- $D\left(p \parallel q\right) \geq 0 \quad \forall\, p, q$
- $D\left(p \parallel q\right) = 0 \Leftrightarrow p = q$
- La divergenza di KL è invariate alle trasformazioni lineari. Siano x e y due vettori aleatori tali che $y = Ax$ con A matrice invertibile, risulta infatti: $D\left(p_x \parallel q_x\right) = \int p_x\left(x\right) \log \frac{p_x(x)}{q_x(x)} dx = \int p_y\left(y\right) \log \frac{p_y(y)}{q_y(y)} dy = D\left(p_y \parallel q_y\right)$

Dimostriamo la prima proprietà: notiamo innanzitutto che la base del logaritmo non è importante in quanto si può passare da una all'altra tramite una costante vale cioè $\log_a x = \frac{\log_b x}{\log_b a}$. Per il logaritmo naturale vale la seguente disuguaglianza: $\ln x \leq x - 1$.

Avremo quindi:

$$D\left(p \parallel q\right) = \int_{-\infty}^{\infty} p\left(x\right) \log \frac{p\left(x\right)}{q\left(x\right)} dx = \\ = -\int_{-\infty}^{\infty} p\left(x\right) \log \frac{q\left(x\right)}{p\left(x\right)} dx$$

Ora

$$\int p\left(x\right) \ln \frac{q\left(x\right)}{p\left(x\right)} dx \geqslant -\int p\left(x\right) \left(\frac{q\left(x\right)}{p\left(x\right)} - 1\right) dx = \\ = -\int q\left(x\right) + \int p\left(x\right) = 0$$

La seconda proprietà è ovvia. La terza discende direttamente dal teorema di trasformazione di vettori alearori.

Andiamo ora a vedere i legami che ci sono tra la divergenza KL e un'altra importante grandezza: l'informazione mutua.

Informazione mutua

Siano x e y due variabili aleatorie, l'informazione mutua tra x e y è definita come:

$$I\left(x, y\right) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p\left(x, y\right) \log \frac{p\left(x, y\right)}{p_x\left(x\right) p_y\left(y\right)} dx dy$$

dove $p(x, y)$ è la ddp congiunta e $p_x(x)$, $p_y(y)$ le ddp marginali. L'informazione mutua ci da un'idea della dipendenza esistente tra due variabili. Più precisamente ci dice quanto la conoscenza di una variabile riduce l'incertezza sull'altra. Questo discorso può essere formalizzato, analizzando il legame tra informazione mutua e entropia differenziale. Avremo infatti:

$$\begin{aligned} I\left(x, y\right) &= H\left(x\right) - H\left(x \left| y\right.\right) = \\ &= H\left(y\right) - H\left(y \left| x\right.\right) = \\ &= H\left(x\right) + H\left(y\right) - H\left(x, y\right) \end{aligned}$$

dove $H(x)$ e $H(y)$ sono le entropie marginali, $H\left(x \left| y\right.\right)$ e $H\left(y \left| x\right.\right)$ sono le entropie condizionate e $H(x, y)$ è l'entropia congiunta. L'entropia è una misura dell'incertezza su una variabile aleatoria. Così $H\left(x \left| y\right.\right)$ da la misura dell'incertezza rimanente nella variabile x una volta data y. Queste relazioni rafforzano l'idea che

l'informazione mutua da una misura della dipendenza tra due variabili aleatorie. Enunciamo anche per l'informazione mutua due proprietà che si dimostrano in maniera simile a quanto fatto per la divergenza KL:

- $I(x,y) \geq 0 \quad \forall x, y$
- $I(x,y) = 0 \Leftrightarrow x$ e y sono indipendenti

Mostriamo ora la relazione esistente tra informazione mutua e divergenza KL. E' immediato verificare che vale:

$$I(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p(x,y) \log \frac{p(x,y)}{p_x(x)\, p_y(y)} dxdy = D(p(x,y) \parallel p_x(x)\, p_y(y))$$

L'estenzione a più variabili è immediata. Vale infatti:

$$I(x_1, x_2, \ldots, x_m) = \int p(x_1, x_2, \ldots, x_m) \log \frac{p(x_1, x_2, \ldots, x_m)}{p_{x_1}(x_1) \ldots p_{x_m}(x_m)} dx_1 \ldots dx_m = \\ = D(p(x_1, x_2, \ldots, x_m) \parallel p_{x_1}(x_1) \ldots p_{x_m}(x_m))$$

Dalle considerazioni fatte si vede che l'informazione mutua è una buona misura per l'indipendenza.

Appendice B

B.1 Matrici di Gram centrate

Vediamo ora come rimuovere l'ipotesi di dati centrati ([32]).

Indichiamo con $\{x_1, ..., x_N\}$ l'insieme delle osservazioni supposte non centrate in H. A partire da queste definiamo $\tilde{\phi}(x_i) = \phi(x_i) - \frac{1}{N}\sum_{n=1}^{N}\phi(x_n)$ le quali risulteranno sicuramente centrate in H. Indichiamo con $\tilde{K}$ la matrice di Gram relativa ai dati centrati. Risulta quindi:

$$\tilde{K}_{ij} = \left\langle \tilde{\phi}(x_i), \tilde{\phi}(x_j) \right\rangle =$$

$$= \left\langle \phi(x_i) - \frac{1}{N}\sum_{m=1}^{N}\phi(x_m), \phi(x_j) - \frac{1}{N}\sum_{n=1}^{N}\phi(x_n) \right\rangle =$$

$$= \left\langle \phi(x_i), \phi(x_j) - \frac{1}{N}\sum_{n=1}^{N}\phi(x_n) \right\rangle - \left\langle \frac{1}{N}\sum_{m=1}^{N}\phi(x_m), \phi(x_j) - \frac{1}{N}\sum_{n=1}^{N}\phi(x_n) \right\rangle =$$

$$= K_{ij} - \frac{1}{N}\sum_{n=1}^{N}K_{in} - \frac{1}{N}\sum_{m=1}^{N}K_{mj} + \frac{1}{N^2}\sum_{m=1}^{N}\sum_{n=1}^{N}K_{mn}$$

Scriviamo quest'ultima espressione in forma matriciale:

$$\begin{aligned}\tilde{K} &= K - \frac{1}{N}\left(\mathbf{1}K + K\mathbf{1}\right) + \frac{1}{N^2}\mathbf{1}K\mathbf{1} = \\ &= K\left(I - \frac{1}{N}\mathbf{1}\right) - \frac{1}{N}1K\left(I - \frac{1}{N}\mathbf{1}\right) = \\ &= \left(K - \frac{1}{N}\mathbf{1}K\right)\left(I - \frac{1}{N}\mathbf{1}\right) = \\ &= \left(I - \frac{1}{N}\mathbf{1}\right)K\left(I - \frac{1}{N}\mathbf{1}\right)\end{aligned} \tag{B.1}$$

Possiamo quindi scrivere che $\tilde{K} = ZKZ$ con $Z = \left(I - \frac{1}{N}\mathbf{1}\right)$.

B.2 Rapporto di determinanti

Diamo qui la dimostrazione di un teorema usato spesso in precedenza.

Teorema B.2.1. *Data una matrice definita positiva partizionata* $\begin{pmatrix} A & B \\ B^T & C \end{pmatrix}$, *vale la seguente relazione:*

$$\begin{aligned}\frac{\det\begin{pmatrix} A & B \\ B^T & C \end{pmatrix}}{\det A \det C} &= \det\begin{pmatrix} I & A^{-1/2}BC^{-1/2} \\ C^{-1/2}B^TA^{-1/2} & I \end{pmatrix} \\ &= \det\left(I - A^{-1/2}BC^{-1}B^TA^{-1/2}\right) \\ &= \prod_i \left(1 - \rho_i^2\right) > 0\end{aligned} \tag{B.2}$$

dove i ρ_i *sono le soluzioni positive del problema generalizzato agli autovalori*

$$\begin{pmatrix} 0 & B \\ B^T & 0 \end{pmatrix} a_i = \rho_i \begin{pmatrix} A & 0 \\ 0 & C \end{pmatrix} a_i$$

Dimostrazione

Dato che $\begin{pmatrix} A & B \\ B^T & C \end{pmatrix}$ è definita positiva, anche A e C lo saranno in quanto sottomatrici di una matrice definita positiva.

Possiamo quindi scrivere:

$$\begin{aligned}
\frac{\det\begin{pmatrix} A & B \\ B^T & C \end{pmatrix}}{\det A \det C} &= \frac{\det\begin{pmatrix} A & B \\ B^T & C \end{pmatrix}}{\det\begin{pmatrix} A & 0 \\ 0 & C \end{pmatrix}} = \\
&= \frac{\det\begin{pmatrix} A & B \\ B^T & C \end{pmatrix}}{\det\left[\begin{pmatrix} A^{-1/2} & 0 \\ 0 & C^{-1/2} \end{pmatrix}\begin{pmatrix} A^{-1/2} & 0 \\ 0 & C^{-1/2} \end{pmatrix}\right]} = \\
&= \det\left[\begin{pmatrix} A^{-1/2} & 0 \\ 0 & C^{-1/2} \end{pmatrix}\begin{pmatrix} A & B \\ B^T & C \end{pmatrix}\begin{pmatrix} A^{-1/2} & 0 \\ 0 & C^{-1/2} \end{pmatrix}\right] = \\
&= \det\begin{pmatrix} I & A^{-1/2}BC^{-1/2} \\ C^{-1/2}B^TA^{-1/2} & I \end{pmatrix} \overset{(a)}{=} \\
&= \det\left(I - A^{-1/2}BC^{-1}B^TA^{-1/2}\right) = \\
&= \det\left(I - BC^{-1}B^TA^{-1}\right) = \\
&= \det\left(I - C^{-1/2}B^TA^{-1}BC^{-1/2}\right) = \\
&= \det\left(I - B^TA^{-1}BC^{-1}\right) \overset{(b)}{=} \prod_i (1-\rho_i) > 0
\end{aligned}$$

Le giustificazioni dei passaggi (a) e (c) si possono trovare in ([31]).

Sempre seguendo ([31] teorema 7.3.7) possiamo scrivere i ρ_i come le soluzioni positive del problema

$$\begin{pmatrix} 0 & A^{-1/2}BC^{-1/2} \\ C^{-1/2}B^TA^{-1/2} & 0 \end{pmatrix} b_i = \rho_i b_i$$

che, con opportuni cambi di variabile possiamo riscrivere come:

$$\begin{pmatrix} 0 & B \\ B^T & 0 \end{pmatrix} a_i = \rho_i \begin{pmatrix} A & 0 \\ 0 & C \end{pmatrix} a_i$$

B.3 Dimostrazione del Lemma 2.1

Enunciamo ora di seguito il Lemma 2.1 dandone poi la dimostrazione.

Lemma 2.1. *Se $P_{\hat{x}\hat{y}}(i,j)$ può essere approssimata come $P_{\hat{x}}(i)\,P_{\hat{y}}(j)\,(1+\varepsilon_{ij})$ con ε_{ij} piccolo (espansione intorno all'indipendenza), allora lo sviluppo di Taylor arrestato al secondo ordine di $I(\hat{x},\hat{y})$ è uguale allo sviluppo di Taylor arrestato al secondo ordine di $I^G(x_G,y_G)$.*

Partiamo scrivendo l'informazione mutua gaussiana nella seguente forma:

$$I^G(x_G,y_G) = -\frac{1}{2}\log\det\left(I-BB^t\right) \quad \text{con } B = D_x^{-1/2}\left(P_{xy}-p_x p_y^t\right)D_y^{-1/2}$$

Assumiamo ora che $(P_{\hat{x}\hat{y}})_{ij} = P_{\hat{x}}(i)\,P_{\hat{y}}(j)\,(1+\varepsilon_{ij})$ quindi $B = D_x^{-1/2}\left(p_x p_y^t + D_x\varepsilon D_y - p_x p_y^t\right)D_y^{-1/2} = D_x^{-1/2}\left(D_x\varepsilon D_y\right)D_y^{-1/2} = D_x^{1/2}\varepsilon D_y^{1/2}$. Dato che ε ha norma piccola, anche B avrà norma piccola. Posso quindi fare la seguente approssimazione:

$$I^G(x_G,y_G) = -\frac{1}{2}\log\det\left(I-BB^t\right) \simeq \frac{1}{2}tr\left(BB^t\right)$$

Ora $tr(BB^t) = tr\left(D_x^{1/2}\varepsilon D_y^{1/2}D_y^{1/2}\varepsilon^t D_x^{1/2}\right) = tr\left(D_x\varepsilon D_y\varepsilon^t\right) = \sum_{i,j}\varepsilon_{ij}^2 P_{\hat{x}}(i)\,P_{\hat{y}}(j)$.
Risulta quindi:

$$I^G(x_G,y_G) \simeq \frac{1}{2}\sum_{i,j}\varepsilon_{ij}^2 P_{\hat{x}}(i)\,P_{\hat{y}}(j)$$

Andiamo ora a espandere $I(\hat{x},\hat{y})$ prendendo il suo sviluppo di Taylor arrestato al secondo ordine.

$$\begin{aligned} I(\hat{x},\hat{y}) &= \sum_{i=1}^{l_x}\sum_{j=1}^{l_y} P_{\hat{x}\hat{y}}(i,j)\log\left(\frac{P_{\hat{x}\hat{y}}(i,j)}{P_{\hat{x}}(i)\,P_{\hat{y}}(j)}\right) \simeq \\ &\simeq \sum_{i,j} P_{\hat{x}}(i)P_{\hat{y}}(j)\,(1+\varepsilon_{ij})\log(1+\varepsilon_{ij}) \simeq \\ &\simeq \sum_{i,j} P_{\hat{x}}(i)P_{\hat{y}}(j)\left(\varepsilon_{ij}+\frac{1}{2}\varepsilon_{ij}^2\right) \end{aligned}$$

dove abbiamo usato l'approssimazione di Taylor per $(1+\varepsilon)\log(1+\varepsilon) \simeq \varepsilon + {}^{\varepsilon^2}\!/_2$. Quindi $I(\hat{x},\hat{y}) \simeq \sum_{i,j} P_{\hat{x}}(i)\,P_{\hat{y}}(j)\,\varepsilon_{ij} + \frac{1}{2}\sum_{i.j} P_{\hat{x}}(i)\,P_{\hat{y}}(j)\varepsilon_{ij}^2$.

La prima sommatoria può essere scritta come $\sum_{i,j} P_{\hat{x}}(i) P_{\hat{y}}(j)(\varepsilon_{ij}+1)-$ $-\sum_{i.j} P_{\hat{x}}(i) P_{\hat{y}}(j) = \sum_{i,j} P_{\hat{x}\hat{y}}(i,j) - 1 = 1 - 1 = 0$. Abbiamo così concluso la dimostrazione. Infatti

$$I(\hat{x},\hat{y}) \simeq \sum_{i,j} P_{\hat{x}}(i) P_{\hat{y}}(i)\varepsilon_{ij}^2$$

■

Appendice C

C.1 Derivata di $F(s)$

Riscriviamo $F(s)$ sviluppando i prodotti:

$$F(s) = (x - As)^t T (x - As) =$$
$$= \left(x^t - s^t A^t\right) T (x - As) = x^t Tx + s^t A^t TAs - x^t TAs - s^t A^t Tx$$

Notiamo che $s^t A^t Tx$ è un numero e che $T^t = T$ in quanto T è l'inversa di una matrice simmetrica (la matrice di covarianza). Possiamo quindi scrivere che:

$$s^t A^t Tx = \left(s^t A^t Tx\right)^t = x^t TAs$$

In definitiva avremo:

$$F(s) = x^t Tx + s^t A^t TAs - 2s^t A^t Tx \tag{C.1}$$

Facendo ora le derivate della (C.1) rispetto alle componenti di s avremo:

$$\nabla F(s) = 2A^t TAs - 2A^t Tx$$

Poniamo ora $\nabla F(s) = 0$. Banalmente avremo:

$$A^t TAs = A^t Tx \tag{C.2}$$

Supponendo ora che A e T siano entrambe invertibili, la soluzione dell'equazione (C.2) esiste unica e vale:

$$s = A^{-1}x$$